

МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ
КИЇВСЬКИЙ НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ ТЕХНОЛОГІЙ ТА
ДИЗАЙНУ

ФАКУЛЬТЕТ МЕХАТРОНИКИ ТА КОМП'ЮТЕРНИХ ТЕХНОЛОГІЙ

КАФЕДРА КОМП'ЮТЕРНИХ НАУК

КВАЛІФІКАЦІЙНА РОБОТА

на тему:

Алгоритмічне та програмне забезпечення для автоматизованого проєктування
головних уборів

Рівень вищої освіти	другий (магістерський)
Спеціальність 122	Комп'ютерні науки
Освітня програма	Комп'ютерні науки

Виконав: студент групи МГІТ1-23

Дмитро ШЕГЕДА

Науковий керівник: д.т.н., проф. **Віктор ЧУПРИНКА**

Рецензент д.ф.-м.н., проф. **Сергій КРАСНИТСЬКИЙ**

Київ 2024

КИЇВСЬКИЙ НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ ТЕХНОЛОГІЙ ТА ДИЗАЙНУ

Факультет мехатроніки та комп'ютерних технологій

Кафедра комп'ютерні науки

Рівень вищої освіти другий (магістерський)

Спеціальність 122 Комп'ютерні науки

Освітня програма Комп'ютерні науки

ЗАТВЕРДЖУЮ

Завідувачка кафедри КН

_____Наталія ЧУПРИНКА

«___»_____2024 року

З А В Д А Н Н Я

НА КВАЛІФІКАЦІЙНУ РОБОТУ СТУДЕНТА

Шегеди Дмитра Сергійовича

1. Тема роботи: Алгоритмічне та програмне забезпечення для автоматизованого проєктування головних уборів.

Науковий керівник роботи д.т.н., професор Чупринка Віктор Іванович, затверджені наказом закладу вищої освіти від 03.09.2023 року, № 188-уч.

2. Вихідні дані до кваліфікаційної роботи: розробка кафедри комп'ютерних наук, літературні джерела з конструювання та технології одягу.

3. Зміст кваліфікаційної роботи (перелік питань, які потрібно розробити)

Вступ, Розділ 1 (Автоматизоване проєктування виробів швейної галузі легкої промисловості); Розділ 2 (Математичне забезпечення для автоматизованого проєктування головних уборів); Розділ 3 (Програмне забезпечення для автоматизованого проєктування головних уборів); Додатки – публікація за темою кваліфікаційної роботи, сертифікати учасника міжнародної наукової конференції та учасника конкурсу наукових робіт студентів.

4. Дата видачі завдання 08.2024р.

КАЛЕНДАРНИЙ ПЛАН

№ з/п	Назва етапів дипломної магістерської роботи	Терміни виконання етапів	Примітка про виконання
1	Вступ		
2	Розділ 1 Автоматизоване проєктування виробів швейної галузі легкої промисловості		
3	Розділ 2. Математичне забезпечення для автоматизованого проєктування головніх уборів		
4	Розділ 3. Програмне забезпечення для автоматизованого проєктування головніх уборів		
5	Висновки		
6	Оформлення кваліфікаційної роботи (чистовий варіант)		
7	Подача кваліфікаційної роботи (проєкту) науковому керівнику для відгуку (за 14 днів дозахисту)		
8	Подача кваліфікаційної роботи (проєкту) для рецензування		
9	Перевірка кваліфікаційної роботи на наявність ознак плагіату		
10	Подання кваліфікаційної роботи на затвердження завідувачу кафедри		

З завданням ознайомлений:

Студент _____ Дмитро ШЕГЕДА
(підпис) (ім'я та прізвище)

Науковий керівник роботи _____ Віктор ЧУПРИНКА
(підпис) (ім'я та прізвище)

АНОТАЦІЯ

Шегеда Д. С. Алгоритмічне та програмне забезпечення для автоматизованого проєктування головних уборів – Рукопис

Дипломна магістерська робота за спеціальністю 122-«Комп'ютерні науки» – Київський національний університет технологій та дизайну, Київ, 2024 рік.

В роботі запропонований методи та алгоритми для автоматизованого проєктування головних уборів. Запропоновані алгоритми реалізовані в програмний продукт для автоматизованого проєктування головних уборів.

Розроблений програмний продукт має дружній інтерфейс та не потребує спеціальних знань з комп'ютерної техніки для роботи з ним.

Ключові слова: математичне та програмне забезпечення, головні убори, швейні вироби.

ABSTRACT

Shcherbatiuk R.V. Algorithmic and software for automated headwear design - Manuscript.

Master's thesis in specialty 122- “Computer Science” - Kyiv National University of Technology and Design, Kyiv, 2024.

The work proposes methods and algorithms for designing hats. The proposed algorithms are implemented in a software product for automated headwear design.

The developed software product has a user-friendly interface and does not require special knowledge of computer technology to work with it.

Keywords: mathematical and software, hats, garments.

ЗМІСТ

1. АВТОМАТИЗОВАНЕ ПРОЕКТУВАННЯ ВИРОБІВ ШВЕЙНОЇ ГАЛУЗІ ЛЕГКОЇ ПРОМИСЛОВОСТІ	8
1.1. Система Автоматизованого Проектування... ..	8
1.2. Плюси і мінуси автоматизації	12
Висновки до першого розділу	15
2. МАТЕМАТИЧНЕ ЗАБЕЗПЕЧЕННЯ ДЛЯ АВТОМАТИЗОВАНОГО ПРОЕКТУВАННЯ ГОЛОВНИХ УБОРІВ	16
2.1. Дизайн-проектування моделей головних уборів	16
2.2. Етапи конструювання головних уборів, методи та алгоритми	18
2.3. Задачі, що виникли при розробці алгоритмів для автоматизованого проектування жіночих головних уборів	35
Висновки до другого розділу.....	38
3. ПРОГРАМНЕ ЗАБЕЗПЕЧЕННЯ ДЛЯ АВТОМАТИЗОВАНОГО ПРОЕКТУВАННЯ ГОЛОВНИХ УБОРІВ.....	39
2.1. Програмні засоби	39
3.2. Структурна схема програмного продукту	40
3.3. Основні функції програмного продукту	41
2.4. Інструкція користувачу по роботі з програмним продуктом для автоматизованого проектування головних уборів	55
Висновки до третього розділу	61
ВИСНОВКИ	62
СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ	63
ДОДАТОК А.....	68
ДОДАТОК Б.....	79
ДОДАТОК В.....	80
ДОДАТОК А Публікація за темою магістерської кваліфікаційної роботи	67
ДОДАТОК Б Сертифікат учасника міжнародної науково-практичної конференції.....	79
ДОДАТОК В Сертифікат учасника конкурсу наукових досліджень студентів	80
ДОДАТОК Г Листинг програмного продукту.....	81

ВСТУП

Згідно з угодою про асоціацію між Європейським Союзом та Україною проблема конкурентоспроможності наших швейних виробів стає особливо гострою. В першу чергу це призведе до необхідності швидше реагувати на запити і вимоги ринку. Вочевидь, що без використання новітніх технологій в процесі проектування одягу не обійтись.

У той же час стрімкий розвиток віртуальних підприємств свідчить про те, що актуальною проблемою стає методичне забезпечення процесів проектування одягу, яке зможе підвищити якість проектування в контексті інформаційного простору та забезпечити доступ вітчизняних дизайнерів до світових ринків інтелектуальних послуг.

Тим паче, що швейна галузь у нашій країні вже готова до сприйняття відповідних новітніх інформаційних технологій. Ще кілька років тому тільки великі підприємства мали змогу встановлювати системи автоматизованого проектування одягу. А нині таку змогу мають і малі підприємства, а навіть і окремі спеціалісти, які виготовляють одяг на індивідуальне замовлення.

В сучасних умовах швидкої зміни моди і смаків споживача та наявності великого асортименту швейних виробів, прискорюється процес морального старіння виробів, тому важливо, щоб темпи їх проектування були якнайшвидшими. Для конкурентоспроможності виробів швейного виробництва і підвищення його ефективності першочергове значення має рівень конструкторсько-технологічної підготовки, якої сприяє застосування систем автоматизованого проектування (САПР). За останній час у нас в країні, як і в усьому світі, здійснюється комп'ютеризація швейного виробництва, активно впроваджуються нові інформаційні технології.

Метою даної кваліфікаційної магістерської роботи є розробка математичного та програмного забезпечення для проектування головних уборів за індивіальними розмірними ознаками голови людини.

Об'єкт дослідження – процес проектування головних уборів.

Методи дослідження. Теоретичні дослідження роботи ґрунтуються на комплексі основних положень виробництва головних уборів та застосування комп'ютерних наук при вирішенні поставленої задачі.

Експериментальні дослідження заключалися в тестуванні розробленого продукту для проектування головних уборів.

Наукова новизна полягає у розробці математичного та програмного забезпечення для проектування проектування головних уборів.

Апробація результатів магістерської роботи. Основні положення і результати магістерської роботи протягом 2023-2024 року були представлені та одержали позитивну оцінку на Всеукраїнському конкурсі студентських наукових робіт та міжнародній науковій конференції: - *III International Scientific and Practical Conference «CURRENT TRENDS IN SCIENTIFIC RESEARCH DEVELOPMENT» 17-19.10.2024 року Бостон, США.*

Публікації. За темою магістерської роботи «Алгоритмічне та програмне забезпечення для автоматизованого проектування головних уборів» опубліковано одна наукових робота: *ВІКТОР ЧУПРИНКА, НАТАЛІЯ ЧУПРИНКА, ДМИТРО ШЕГЕДА ПРОГРАМНИЙ ПРОДУКТ ДЛЯ АВТОМАТИЗОВАНОГО ПРОЄКТУВАННЯ ГОЛОВНИХ УБОРІВ/ Proceedings of III International Scientific and Practical Conference «CURRENT TRENDS IN SCIENTIFIC RESEARCH DEVELOPMENT» 17-19.10.2024 року Бостон, США, с.152-160 URL: <https://sci-conf.com.ua/iii-mizhнародna-naukovo-praktichna-konferentsiya-currenttrends-in-scientific-research-development-17-19-10-2024-boston-ssha-arhiv/>*

Структура та обсяг магістерської роботи. Магістерська робота містить вступ, три розділи із висновками, загальні висновки досліджень, список використаних джерел, додатки. Загальний обсяг магістерської роботи становить 80 сторінка, у тому числі 29 рисунків, 20 формул. Список використаних джерел становить 33 позицій. Додатки займають обсяг 11 сторінок.

РОЗДІЛ 1. АВТОМАТИЗОВАНЕ ПРОЕКТУВАННЯ ВИРОБІВ ШВЕЙНОЇ ГАЛУЗІ ЛЕГКОЇ ПРОМИСЛОВОСТІ

1.1. Система Автоматизованого Проектування

САПР – це спеціальний програмний продукт, що дозволяє прискорити і полегшити розробку моделей одягу.

Завдання САПР[1-3]:

- скорочення затрат та часу на проектування одягу;
- підвищення якості розроблених моделей;
- швидке реагування на появу нових ідей на ринку попиту;
- автоматизована підтримка технологічної підготовки при виготовленні серійного, малосерійного та індивідуального одягу за різними розмірними ознаками;
- накопичення та широке застосування розробок;
- повний перехід від ручного конструювання та вводу лекал з дигитайзера на виконання усіх етапів безпосередньо на персональному комп'ютері.

Функціонування САПР визначає нову прогресивну технологію проектування, що заснована на взаємодії людини та ЄОМ.

В 50-х минулого ст. вже існували перші зразки устаткування САПР. Уперше аббревіатуру САПР використав основоположник цього напрямку Айвен Сазерленд. На початку 60-х років ним були прочитані лекції в Масачусетському технологічному університеті. Починаючи з 70-х формуються передумови створення САПР, теоретичні положення і перші алгоритми. САПР забезпечує проектувальника могутнім інструментом для: перетворення контурів і побудови конструкцій, можливостями здійснення поворотів, перенесень, симетричних відображень, масштабування, компактним зберіганням уніфікованих варіантів конструкцій і стандартних деталей.

В даний час у швейній промисловості використовується значне число систем автоматизованого проектування, найбільш відомими з яких є АССОЛЬ, ГРАЦІЯ, КОМТЕНС, ЛЕКО, GERBER, LECTRA (INVESTRONICA), GRAFIS та ряд інших[4-7].

Фірмою ***Vilar*** розроблена комп'ютерна методика побудови рукава за параметрами пройми; побудова рукава-реглан із широким діапазоном регулювання форми рукавів. Система пропонує 50 основ і 150 моделей. Будь-яка побудова допускає внесення виправлень. Лекала можуть виводитися на принтер (A4, A3, A2) або плотер. При друці деталі виводяться в роздріб із вказівкою ліній з'єднання. Ця система не дорога, не вимагає великих площ при встановленні.

Деякі системи, такі як «**Ассоль**» пропонують готові методики побудови креслень БК, надаючи користувачеві можливість модифікації за допомогою зміни їх параметрів. Часткова параметризація процесу конструювання може відображатися у можливості запису послідовності дій (макросів), яка часто повторюється, як, наприклад, у САПР «Ассоль», так і в реалізації процесу «надання розміру» непараметричного креслення, як, наприклад, у САПР «Optitex».

Комплекс програм ***JULIVI*** (м. Луганськ) для автоматизації майже всіх процесів швейного виробництва: від проектування моделей до управління виробництвом.

Всі згадані системи пропонують конструктору практично однакові можливості. Справа за фахівцем – яку систему вибрати.

Проведений аналіз показує, що не дивлячись на описані значні переваги САПР, залишається не до кінця вирішеною низка питань. Наприклад, у комп'ютері практично неможливо врахувати фізико-механічні параметри тканини; значну вартість обладнання, яке необхідно для підтримки продуктивної роботи більшості описаних САПР; трудомісткість введення старих лекал у правильну комп'ютерну форму.

Все вище сказане зумовлює необхідність подальшого вивчення алгоритмів реалізації існуючих САПР одягу з метою вдосконалення та спрощення. Необхідно також відзначити, що вітчизняні системи починають складати конкуренцію зарубіжним розробкам.

Кожна система автоматизованого проектування заслуговує на те, щоб бути використаною в системі підготовки майбутніх вчителів технологій. Але для повного навчання потрібно обирати системи з повним комплектом операцій, починаючи з ескізного розрахунку й закінчуючи автоматичною розкладкою лекал, з урахуванням всіх характеристик використовуваної тканини.

Багато сучасних САПР швейних виробів на додаток до засобів автоматизації праці конструктора пропонують можливості автоматизованого формування технічних і художніх ескізів.

САПР (Система Автоматизованого Проектування) — це програмне забезпечення, яке використовується для розробки, моделювання та оптимізації проектів у різних галузях, таких як інженерія, архітектура, машинобудування, електроніка тощо. З допомогою САПР фахівці можуть створювати двовимірні (2D) та тривимірні (3D) моделі об'єктів, проводити аналіз конструкцій, симуляцію їхньої поведінки та перевіряти на відповідність технічним вимогам.

У сучасному виробництві широке поширення одержали системи автоматизованого проектування (САПР, computer aided design), які дозволяють проектувати технологічні процеси з меншими витратами часу та засобів, зі збільшенням точності спроектованих процесів і програм обробки, що скорочує витрати матеріалів та час обробки, завдяки тому, що режими обробки також розраховуються та оптимізуються за допомогою ЕОМ.

Технічне забезпечення САПР засновано на використанні обчислювальних мереж і телекомунікаційних технологій, персональних комп'ютерів та робочих станцій.

Математичне забезпечення САПР характеризується різноманітністю методів обчислювальної математики, статистики, математичного програмування, дискретної математики, штучного інтелекту. Програмні комплекси САПР відносяться до числа найбільш складних сучасних програмних систем, заснованих на операційних системах Unix, Windows, мовах програмування C, C++, Java і інших, сучасних CASE технологіях, реляційних і об'єктно-

орієнтованих системах керування базами даних (СКБД), стандартах відкритих систем і обміну даними в комп'ютерних середовищах.

Проектування, при якому всі проектні рішення або їхня частина одержують шляхом взаємодії людини та ЕОМ, називають автоматизованими на відміну від ручного (без використання ЕОМ) або автоматичного (без участі людини на проміжних етапах). Система, що реалізує автоматизоване проектування, являє собою систему автоматизованого проектування (в англomовному написанні **CAD** System – Computer Aided Design System). САПР (або CAD) звичайно використовуються разом із системами автоматизації інженерних розрахунків і аналізу **CAE** (Computer-Aided engineering). Дані із CAD- систем передаються в **CAM** (Computer-Aided manufacturing) – систему автоматизованої розробки програм обробки деталей для верстатів.

CAE – автоматизоване конструювання, використання спеціального програмного забезпечення для проведення інженерного аналізу міцності та інших технічних характеристик компонентів, виконаних у системах автоматизованого проектування. Програми автоматизованого конструювання дозволяють здійснювати динамічне моделювання, перевірку та оптимізацію виробів і засобів їхнього виробництва.

CAM – автоматизоване виробництво. Термін використовується для позначення програмного забезпечення, основною метою якого є створення програм для керування верстатами зі ЧПК (числове програмне керування). Вхідними даними САМ- системи є геометрична модель виробу, розроблена в системі автоматизованого проектування. У процесі інтерактивної роботи із тривимірною моделлю в САМ системі інженер визначає траєкторії руху різального інструменту по заготівлі виробу, які потім автоматично верифікуються, візуалізуються (для візуальної перевірки коректності) і обробляються постпроцесором для одержання програми керування конкретним верстатом.

Структура САПР. САПР складається з проектуючої і обслуговуючої підсистем. Проектуючі підсистеми безпосередньо виконують проектні

процедури. Прикладами проектуючи підсистем можуть слугувати підсистеми геометричного тривимірного моделювання механічних об'єктів, виготовлення конструкторської документації, схемотехнічного аналізу, трасування з'єднань у друкованих платах.

Обслуговуючі підсистеми забезпечують функціонування проектуючи підсистем, їхню сукупність часто називають системним середовищем (або оболонкою) САПР. Типовими обслуговуючими підсистемами є підсистеми керування проектними даними (PDM – Product Data Management), керування процесом проектування (DesPM – Design Process Management), користувацького інтерфейсу для зв'язку розробників з EOM, CASE (Computer Aided Software Engineering) для розробки та супроводу програмного забезпечення САПР, навчальні підсистеми для освоєння користувачами технологій, реалізованих у САПР.

На сьогодні створено велику кількість програмно-методичних комплексів для САПР із різними ступенем спеціалізації й прикладною орієнтацією. У результаті автоматизація проектування стала необхідною складовою частиною підготовки інженерів різних спеціальностей; інженер, що не володіє знаннями та не вміє працювати в САПР, не може вважатися повноцінним фахівцем.

1.2. Плюси і мінуси автоматизації

Сьогодні світ зіткнувся з такими нагальними питаннями як автоматизація, звільнення та скорочення. Це торкнулося не тільки IT-індустрії, а й інших сфер, включаючи фармацевтику, автомобільну та інші галузі промисловості. Виробники одягу стали єдиним винятком, яке не захлеснула цифрова революція. В результаті замість того, щоб сконцентруватися на процесі виробництва одягу, виробники приділяють все більше уваги електронної комерції і маркетинговим стратегіям[9-11].

Саме поняття автоматизація передбачає розробку особливих систем, здатних замінити людський ресурс при виконанні певних завдань.

У 2017 році McKinsey Global Institute надав свої прогнози щодо автоматизації. Згідно зі звітом, до 2055 року більшість завдань, виконуваних

людиною, буде автоматизовано починаючи від фізичної праці і закінчуючи збором/обробкою даних. Процес автоматизації торкнеться як самого виробництва одягу, так і роздрібної торгівлі. В результаті вся модна індустрія зазнає колосальні зміни.

Актуальні проблеми автоматизації швейного цеху.

Обладнання та доставка речей масового споживання буде коштувати досить дорого. Все це вкупі зі зростаючими цінами на робочу силу є значною перешкодою, тому рідкісний виробник націлений на повну автоматизацію. Навіть такі гіганти виробництва як Китай, намагаючись йти в ногу з часом, звертаються за пошуком робочої сили в Ефіопії, Бангладеш і інші країни, працюючи за системою аутсорсингу.

Серед інших нагальних проблем також виділяють:

- розробку стилів;
- швидкість виконання;
- терміни доставки;
- забезпечення стабільної якості;
- гнучкість.

Крім того, перед виробниками стоїть складне завдання з виробництва одягу для країн з низьким рівнем заробітної плати.

Переваги автоматизації. Автоматизація це не тільки можливість прискорити виробничий процес. Це також відмінний спосіб стати більш конкурентоспроможним на ринку. Хоча сьогодні більшість робіт здійснюється вручну, інвестиції в автоматизацію життєво необхідні для того, щоб відповідати світовим стандартам. Полювання за дешевою робочою силою не може тривати вічно, а єдиний вихід із замкнутого кола нові технології.

Серед інших важливих переваг автоматизації також виділяють:

- енергозбереження;
- зниження витрат на техобслуговування;
- зниження витрат на заміну запчастин;
- зменшення кількості необхідного простору;

- спрощення управління;
- підвищений контроль якості;
- низьку ймовірність браку;
- спрощену звітність і т. д.

Існують і інші причини переходу до автоматизації в швейній індустрії. Виробництво одягу спочатку прийшло з країн з високим рівнем життя та заробітними платами, таких як США і країни Європи. Виробництво перемістилося в бідні країни. В результаті бренди пішли від творчості до маркетингу, в той час як виробництва стало приділятися все менше уваги. І якщо раніше процес знаходився під контролем досвідчених фахівців, які знають свою справу, то тепер доля розвитку швейного виробництва опинилася в руках у виробників-контрактників.

Останні ж прагнуть до підвищення мінімальних обсягів замовлень та усунення множинних модифікації дизайну одягу для оптимізації виробництва. У результаті елемент творчості практично зник з швейній індустрії, а помилка в прогнозуванні модних тенденцій здатна призвести до значних фінансових втрат.

Швейні боти в легкій промисловості. Швейні боти являють собою автоматизовані швейні машини, що дозволяють знизити потребу в людському ресурсі. Крім того, вони здатні:

- знизити виробничі витрати;
- підвищити якість виробництва;
- скоротити терміни виконання;
- повернути торговим маркам контроль над процесом виробництва.

Такі машини можуть виявитися відмінним виходом в гонитві за здешевленням виробництва, доставки, створенням більш якісного товару. Їх ефективність і надійність набагато вище людської праці. Швидкість виконання залишиться такою ж, як і у людини, однак на відміну від людей, роботи рідко помиляються і не потребують в перервах на обід. Це відносно небагато з того, що швейні боти здатні зробити для швейної індустрії.

Повна автоматизація виробництва процес досить тривалий. Це відбудеться не відразу, проте вже сьогодні ми можемо спостерігати частковий перехід до автоматизованого виробництва на наступних етапах виготовлення:

- виробництва тканини;
- фарбування;
- оздоблення.

Людський ресурс все ще використовується. Доцільний він тоді, коли людина може управляти машиною для досягнення більшої продуктивності або контролюючи якість. У результаті, витрати на виробництво можуть стати мінімальними.

На сьогоднішній день технологічні новатори прагнуть удосконалити швейні машини, додаючи нові операції, а також можливості роботи з різними типами тканин. Перш ніж роботи остаточно "захоплять" виробництво, потрібно подолати масу перешкод.

Зворотна сторона автоматизації. Існує і інша сторона процесу автоматизації. Зрозуміло, роботизирование може стати вирішенням багатьох проблем. Багато хто також вважають, що автоматизація здатна позбавити людей в бідних країнах від жалюгідних умов праці. Але яка альтернатива є у тих, хто утримує свої сім'ї і втратить роботу в результаті автоматизації? Таким чином, останнє питання, яке потребує відповіді, полягає в тому, чи здатна парочка робітників звільнити людей від страждань, принесених безробіттям? Варто плодити голодних і бездомних, що неодмінно станеться у разі повної автоматизації?

Висновки до першого розділу

1. В результаті замість того, щоб сконцентруватися на процесі виробництва одягу, виробники приділяють все більше уваги електронної комерції і маркетинговим стратегіям
2. Впровадження комп'ютерних технологій у проєктування та виробництво одягу на малих підприємствах, які працюють за індивідуальними заказами, є актуальною проблемою

РОЗДІЛ 2. МАТЕМАТИЧНЕ ЗАБЕЗПЕЧЕННЯ ДЛЯ АВТОМАТИЗОВАНОГО ПРОЄКТУВАННЯ ГОЛОВНИХ УБОРІВ

2.1. Дизайн-проектування моделей головних уборів

На сучасному етапі дизайн-проектування моделей головних уборів слід враховувати не тільки модні тенденції, крій, силует, характеристики матеріалу, а також форму обличчя споживача. Адже обличчя завжди було ключовою частиною зовнішності. Його форма та риси впливають на враження, на образ в цілому. Тому при правильному підборі головного убору можна сформувати власний стиль у відповідності з даними природою особливостями зовнішності[12-17].

Обличчя містить найбільшу кількість специфічних ознак, що притаманні конкретній людині. Форма лицьової частини голови у фас та профіль є важливими відомостями для розробки сучасних моделей головних уборів. За морфологічними ознаками виділяють сім поширених форм лицьової частини голови : з них чотири основних (овальна, трикутна, кругла, квадратна), а також підтипи – змішані види (ромбоподібна, прямокутна, грушоподібна)[18-23] (рис.2.1).

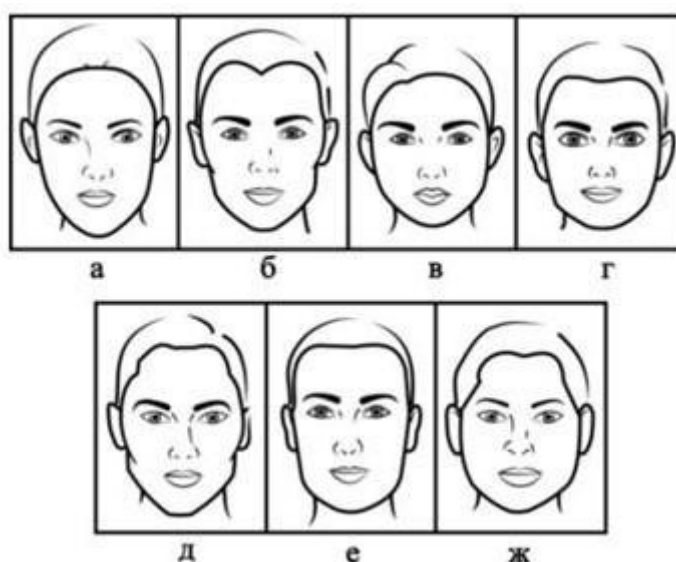


Рис. 2.1. Форми лицьової частини голови людини: а – овальна, б – трикутна, в – кругла, г – квадратна, д -ромбоподібна, е – прямокутна, ж - грушоподібна

Щоб визначити особливості обличчя споживача, слід визначити його пропорційність, а саме: як відноситься вертикальна та горизонтальна частини. Тобто, як відноситься висота обличчя (відстань від крайової лінії росту волосся по середній лінії обличчя до підборіддя) до ширини обличчя в найширшому місці. Ідеальним є відношення $V/Ш=3/2$, де V – висота обличчя, $Ш$ – ширина обличчя в найширшому місці (рис. 2.2).

Форму обличчя визначають по чотирьом основним лініям: висоти, ширини лоба, ширини вилиць та ширини підборіддя .

Форма обличчя визначається за такими вимірами (рис. 2.3):

- висота (V) обличчя визначається відстанню від крайової лінії росту волосся по середній лінії обличчя до підборіддя;
- ширина лоба ($Шл$) визначається відстанню між крайовими лініями росту волосся від правої скроні до лівої через середину лоба;
- ширина вилиць ($Шв$) визначається відстанню між вилицями в найширшому місці;
- ширина підборіддя ($Шп$) визначається відстанню на рівні 1 см від й

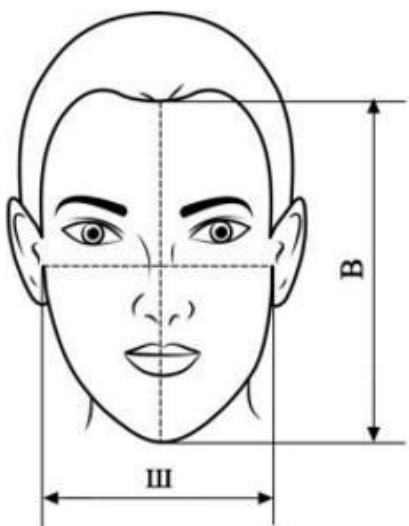


Рис. 2.2. Відношення висоти до обличчя

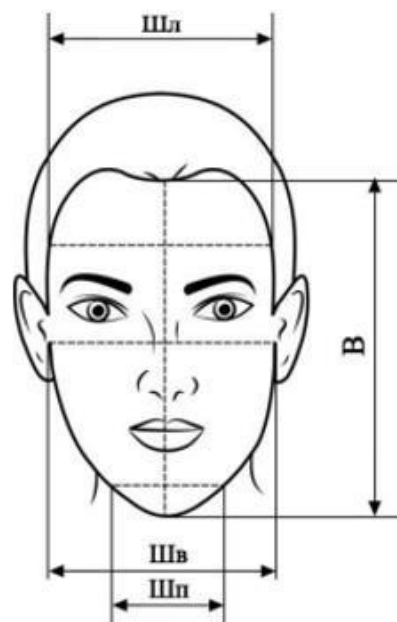


Рис. 2.3. Пропорційне зображення обличчя з позначенням лінійних вимірів

У класифікації Ритвінської Л. Б. (рис. 2.4) надані морфологічні ознаки голови людини, конструктивні пояси та лінії композиційного устрою (стабільні та мобільні). Все це дає змогу визначити узгодженість ліній головного убору з лініями обличчя.

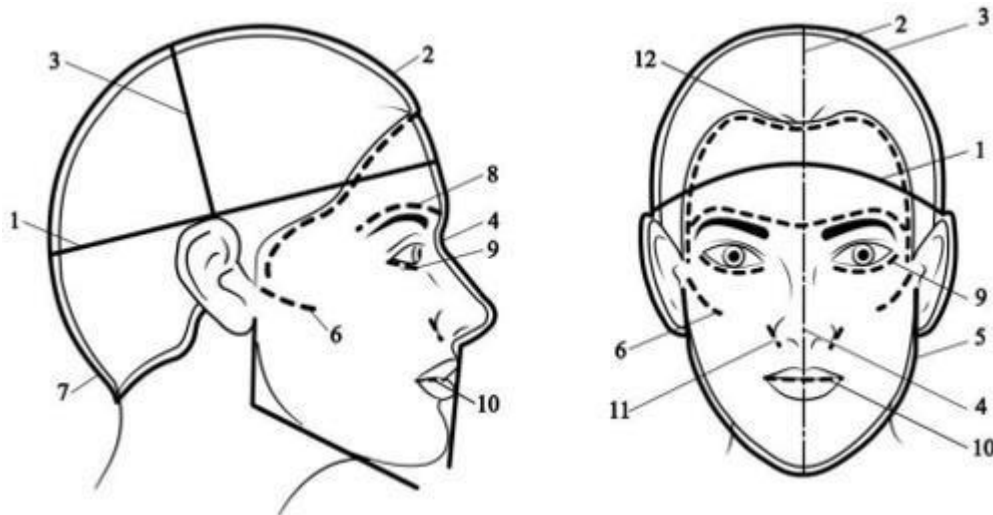


Рис. 2.4. Класифікація морфологічних ознак голови людини: конструктивні пояси (1 – обхват голови, 2 – повздовжня лінія, 3 – поперечна лінія); лінії композиційного устрою (стабільні лінії: 4 – профільна лінія, 5 – лінія овалу обличчя, 6 – лінія вилиць та виличних дуг, 7 – лінія потилиці; мобільні лінії: 8 – лінія брів, 9 – лінія розрізу очей, 10 – лінія рота, 11 – лінія крил та кінчика носа, 12 – лінія краю волосяного покриву голови)

Важливим є гармонійне узгодження об'єму та форми головного убору з формою обличчя, лінії головних уборів узгоджуються з лініями брів, розрізу очей та взаємодіють з іншими рисами обличчя. Наприклад, лінія піднятої вверх криси капелюха і лінія форми головки узгоджуються з загальною формою голови та обличчя.

2.2. Етапи конструювання головних уборів, методи та алгоритми

Жіноча шляпа з широкими полями (n-клинна).

Алгоритм побудови n-клинної шляпи з широкими полями (рис. 2.5):

1. Розраховуємо кут сектору $\alpha = \frac{2\pi}{n}$, де n – кількість клинів;
2. Будуємо сектор радіусом $(R + \Delta) < \alpha$, де R – глибина голови.

Проводимо дугу радіусом $\frac{R}{2}$ для цього сектору та знаходимо точки перетину дуги радіусом $\frac{R}{2}$ із сторонами сектора;

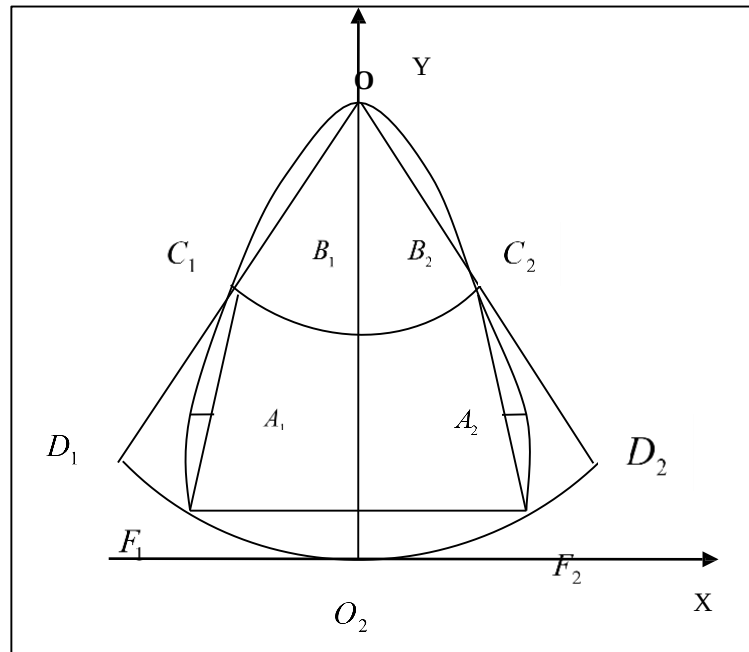


Рис. 2.5. Побудова основного контура клина

3. Визначимо довжину одного клина: $l = \frac{L + \Delta \cdot n + \Delta_1}{n}$, де L - окружність голови, $\Delta = 16$, Δ_1 - припуск на посадку (10мм);
4. Робимо засічки від точки O_2 радіусом $\frac{l}{2}$ у дві сторони, утворилися точки F_1 та F_2 ;
5. Сполучаємо точки C_1 та F_1 і C_2 , F_2 ;
6. Знаходимо точку середини відрізків C_1F_1 , C_2F_2 , C_1O , C_2O ;
7. Від точок A_1 та A_2 відкладаємо перпендикуляр довжиною $\frac{R}{20}$, а від точок B_1 та B_2 довжиною $\frac{R}{40}$;
8. З'єднуємо точки F_1, C_1, O, C_2, F_2 .

Побудова клина шляпи завершено.

9. Побудуємо поле шляпи. Для цього необхідно побудувати два півкола – одне з радіусом R , інше – з радіусом $R + h$, де h (ширина поля шляпи) вводиться програмно (рис. 2.6).

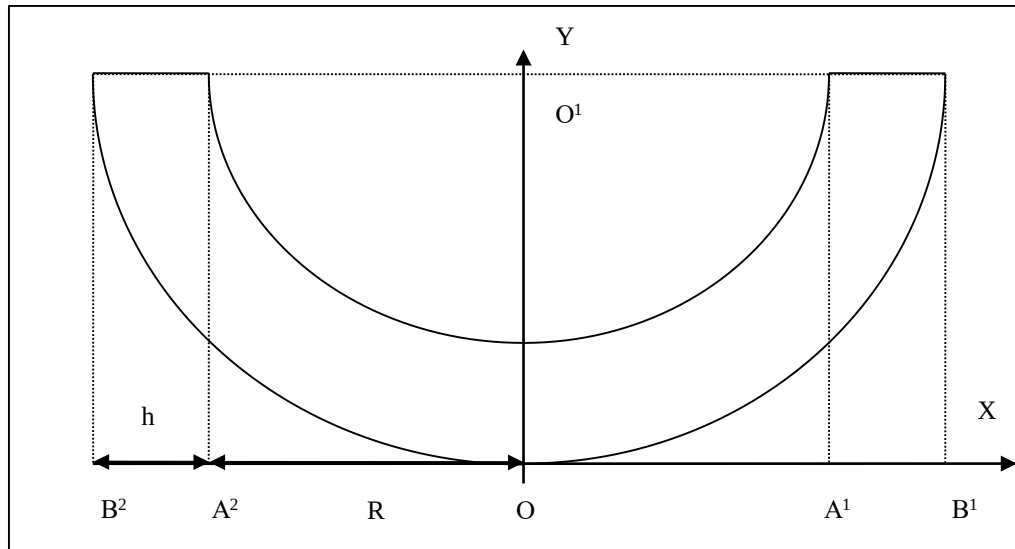


Рис. 2.6. Поле шляпи

Отже, лекала для пошиву п-клинної шляпи з широкими полями готові.

Молодіжна панама.

Молодіжна панама складається з головки округлої форми та опущених нешироких полів. Головка в свою чергу складається з п'яти деталей – середньої подовженої та чотирьох невеликих клинів неправильної форми, які створюють бокові стінки панам. Поле панам також складається з двох частин. В шви бокових стінок панам вставляють вистрочені вузькі пояски і скріплюють їх декоративними пряжками. Потім прострочують панаму рельєфним декоративним швом і з'єднують з полями.

Алгоритм побудови клина, неправильної форми:

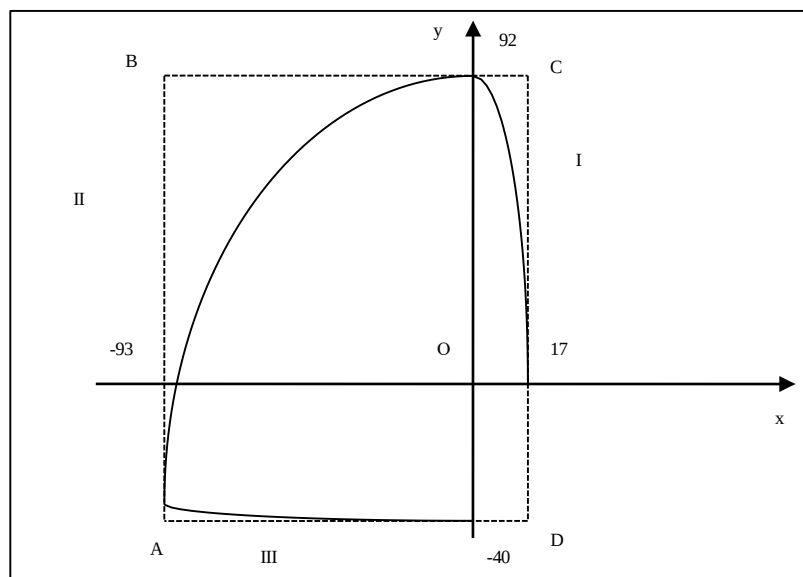


Рис. 2.7. Клин молодіжної панам

Знайдемо координати точок A(-93;-45), B(-93;92), C(17;92), D(17;-40) та побудуємо прямокутник ABCD (рис.2.7). Потім необхідно побудувати три дуги – чверті еліпсів областях:

Координати точок показані на рис. 2.7.

Програмно це реалізовано за допомогою наступного алгоритму[24-25]:

Початок

$$\varphi = \frac{\varphi_2 - \varphi_1}{n};$$

Для $i = 0$ до n Робити

Початок

$$xr[i] = xc + a * \cos(\varphi_1 + \varphi * i);$$

$$yr[i] = yc + b * \sin(\varphi_1 + \varphi * i);$$

Кінець

Кінець

Де задаються xc, yc – координати центра еліпса; a, b – висота осей еліпса;

$Fi1, Fi2$ – область а якій знаходиться еліпс.

Побудова середньої деталі панам:

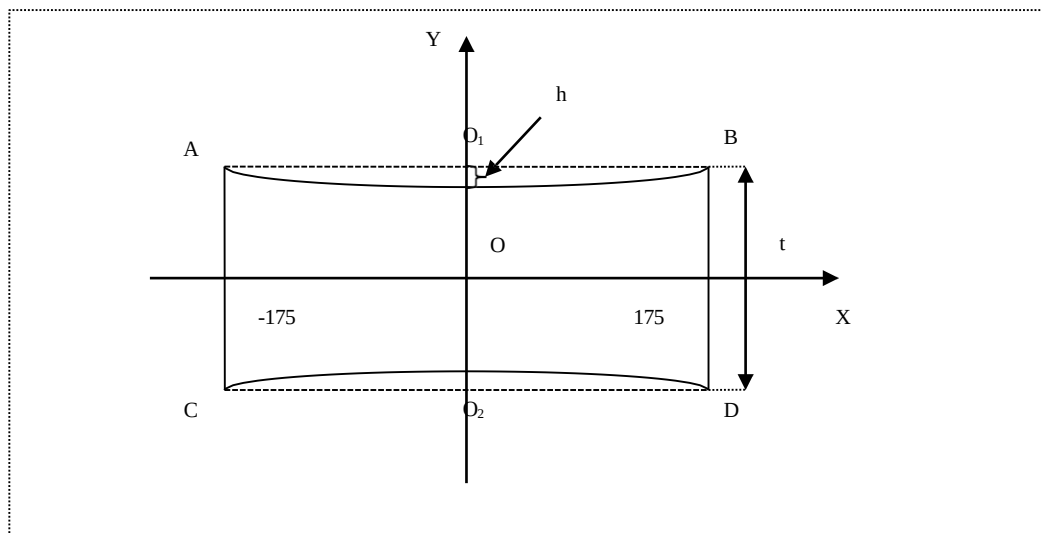


Рис. 2.8. Середня деталь верху панам

Для побудови цієї деталі необхідно спочатку побудувати прямокутник з довжиною 350мм і шириною t . Потім будуюмо два пів еліпси, з центрами у точках $O_1(0, t/2)$ і $O_2(0, -t/2)$, мала піввісь еліпса становить h , а велика – 175мм. Висоту деталі (t) знаходимо за формулою:

$$t = GlGol + \Delta - 2 * H;$$

де $GlGol$ – глибина голови, вказується при побудові лекала; Δ - припуски на шви; H – висота клина панам; h – константа, розраховується відповідно до розмірів клина панам і становить в нашому випадку 7мм.

Довжина AB і CD – розраховується відповідно до суми двох довжин еліпса (II) при побудові клина, і в нашому випадку становить 150мм.

Побудова поля молодіжної панам:

Для побудови поля панам побудуємо розгортку зрізаного конуса.

Для цього знайдемо R , l , φ : $R = \frac{ObhGol + \Delta}{2\pi}$; де $ObhGol$ - обхват голови,

вказується при побудові лекала;

$$l = \frac{R}{\sin(\alpha/2)} \text{ і } \varphi = \frac{2\pi * R}{l}, \text{ де } \varphi - \text{кут розгортки.}$$

В результаті, після проведення всіх побудов, отримаємо поле на рис.2.9, h – ширина поля, вказується програмно і залежить від побажання користувача.

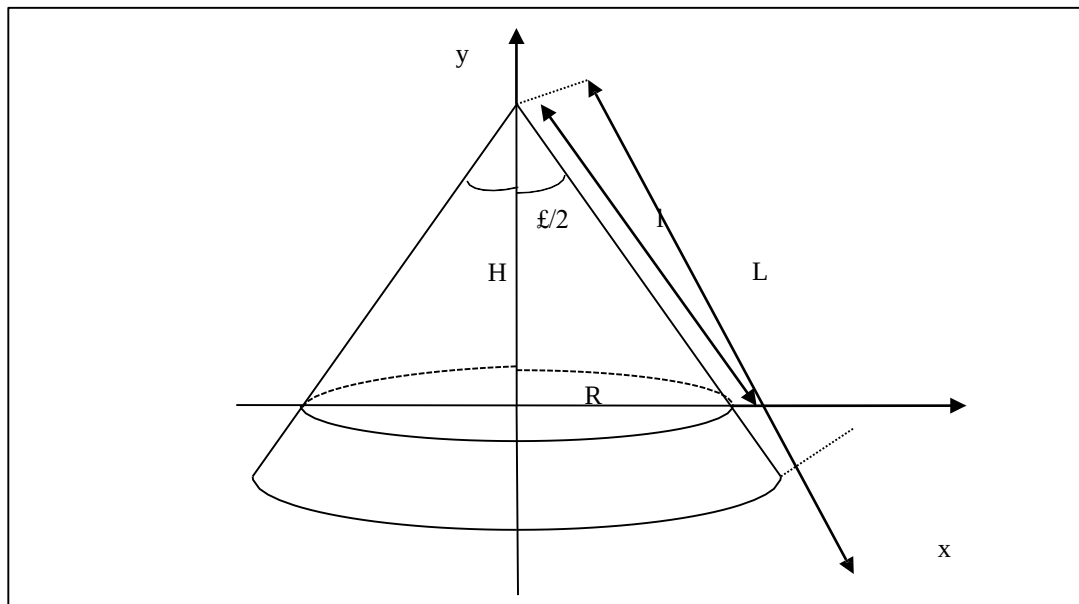


Рис. 2.9. Розгортка конуса для побудови поля панам

Пошук радіуса, довжини і кута дуги реалізовано за допомогою наступного алгоритму:

Початок

$$R = \frac{ObhGol * \pi}{2};$$

$$l = \frac{R}{\sin(Alf / 2)};$$

$$\varphi = \frac{\pi * R}{l};$$

Кінець

Потім реалізуємо побудову самого поля, це показано в наступному алгоритмі:

Початок

$$\varphi_0 = \frac{-\pi}{2} + \frac{\varphi}{2};$$

$$Delta\varphi = \frac{\varphi}{18};$$

Для i=0 до 18 Робити

Початок

$$Xpol[i] = (L + H) * \cos(\varphi_0 - i * Delta\varphi);$$

$$Ypol[i] = (L + H) * \sin(\varphi_0 - i * Delta\varphi);$$

Кінець

$$\varphi_0 = -\frac{\pi}{2} - \frac{\varphi}{2};$$

Для i=0 до 18 Робити

$$Xpol[i+19] = L * \cos(\varphi_0 - i * Delta\varphi);$$

$$Ypol[i+19] = L * \sin(\varphi_0 - i * Delta\varphi);$$

Кінець

$$Xpol[38] = Xpol[0];$$

$$Ypol[38] = Ypol[0];$$

$$Npol = 39;$$

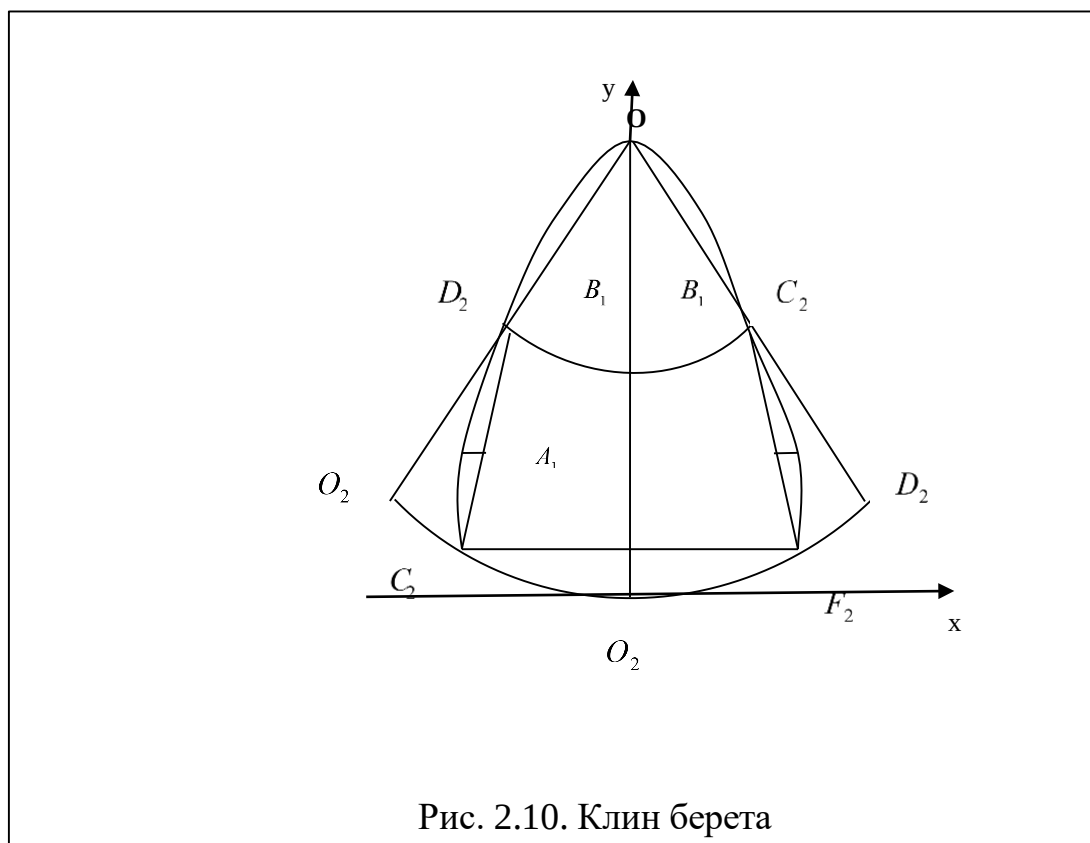
Кінець

де X_{pol} , Y_{pol} - координати відповідних точок фігури, N_{pol} - кількість точок фігури, яка використовується при побудові.

Берет зі шнурівкою.

Алгоритм побудови n -клинного берету.

Берет з шнуровкою складається з звичайних клинів, і клина з шнуровкою. Для побудови стандартного клина використаємо алгоритм, описаний в простому береті. Для нього визначимо параметри, сектор клина становить радіус $(R + L + \Delta)$, де R - глибина голови, $L = 100\text{мм}$ – подовження клина для моделювання берету.



А також необхідно побудувати n -й клин для шнуровки берета. Він буде в два рази ширший за попередній, тобто використаємо конструкцію, що показана на рис. 2.10. і вдосконалимо її.

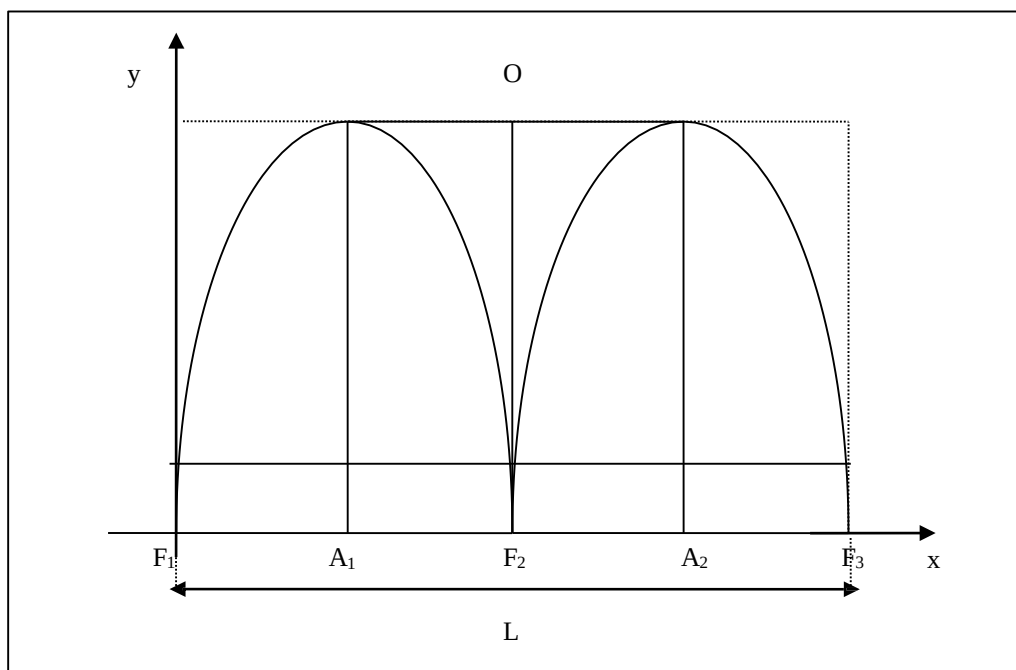


Рис.2.11.

Довжина $L=2 \cdot F_1F_2$.

Поділимо відрізок A_1A_2 пополам, і проведемо OF_2 .

Потім поділимо кожний з відрізків A_1F_2 і F_2A_2 ще пополам, і сполучимо їх променями так як показано на рис. 7, $h=100$ мм.

А також зробимо по 6 блочок діаметром в 10мм на кожній стороні ребра, див. рис. 2.8.

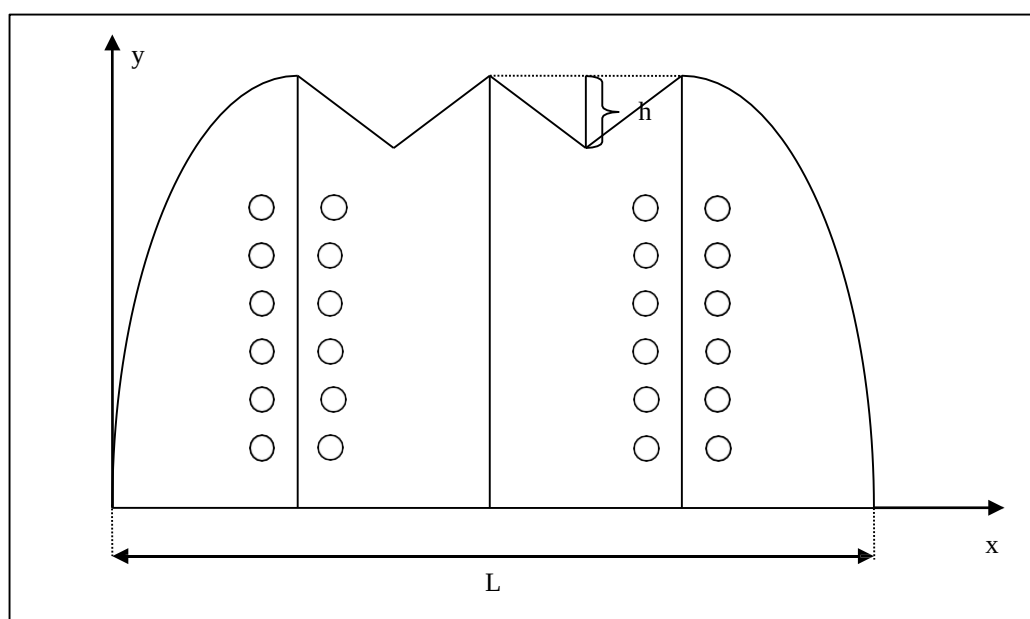


Рис 2.12. Клин берета з шнуровкою

Залишилося побудувати околиш берета, довжиною в обхват голови з припуском на шов і шириною $H=100\text{мм}$.

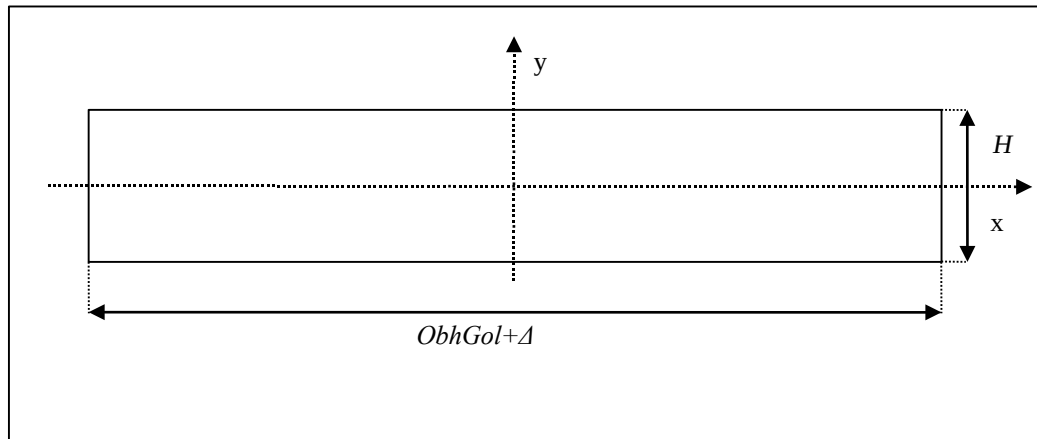


Рис. 2.13. Околиш берета

Конфедератка

Алгоритм побудови конфедератки наступний:

1. Побудуємо верхню частину конфедератки. Для цього необхідно враховувати обхват голови і сам розмір верхньої частини шапки (квадрата), який буде однаковим для всіх розмірів конфедератки і становитиме $H=340\text{ мм}$. Для економії матеріалу зробимо чотири заокруглення на кутах квадрата, з проектуємо чотири дуги, радіус яких буде розраховуватися за формулою $r=2*(ObhGol/4 + \Delta)/\pi$. Програмно побудова реалізується за допомогою алгоритму побудови дуги[27]:

Початок

$$D_alfa = AlfaE - AlfaB;$$

$$n = 18;$$

Для $i=0$ до n Робити

Початок

$$Xd[i] = R * \cos(AlfaB + D_alfa * i / n) + X_0;$$

$$Yd[i] = R * \sin(AlfaB + D_alfa * i / n) + Y_0;$$

Кінець

$$n = n + 1;$$

Кінець

Де $X0, Y0$ – початкова точка, R – радіус дуги, $\text{AlfaB}, \text{AlfaE}$ – кути.

В результаті отримаємо фігуру, див. рис. 2.14:

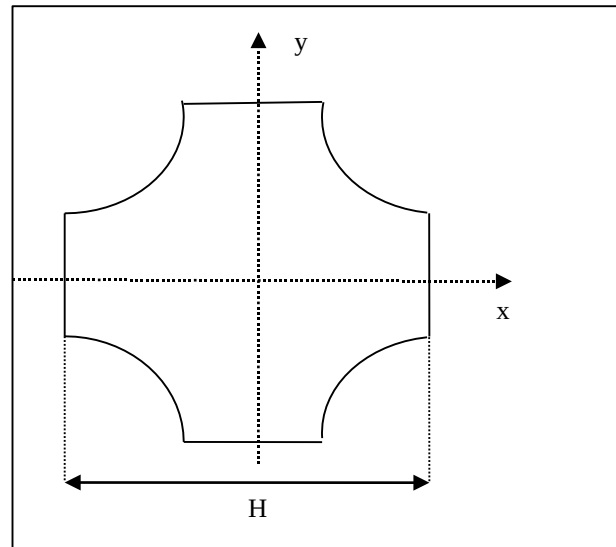


Рис. 2.14 Верхня частина конфедератки

- Тепер побудуємо околицю конфедератки, як в попередній шляпі на рис.2.14. Його ширина дорівнюватиме обхвату голови з припуском на шов, а висота 250мм.

Ковбойська шляпа

Ковбойська шляпа складається з трьох основних частин: поле, верх шляпи і дінце.

Поле шляпи будується за допомогою двох еліпсів – внутрішнього і зовнішнього. Для побудови внутрішнього еліпса, необхідно розрахувати його велику і малу піввісь. Для цього використаємо рівняння: $R = \frac{a_g + b_g}{2}$, де a_g - мала піввісь еліпса, b_g - велика піввісь еліпса. З системи рівнянь:

$$\begin{cases} \frac{ObhGol}{2\pi} = \frac{a_g + b_g}{2} \\ k = \frac{b_g}{a_g} = 1,5 \end{cases}$$

розрахуємо $a_g = \frac{ObhGol}{(1+k)\pi}$; $b_g = a_g \cdot k$.

Для побудови зовнішнього еліпса, значення великої і малої піввісі візьмемо з форми при введенні розмірних ознак шляпи.

Алгоритм для побудови поля шляпи:

Початок

$$B_g = \frac{ObhGol * (1 + k)}{pi};$$

$$A_g = k * B_g;$$

$$Npole_vnutr = 37;$$

Для $i=0$ до 36

Робити

Початок

$$Xpole_vnutr[i] = A_g * \cos\left(\frac{pi * i}{18}\right);$$

$$Ypole_vnutr[i] = A_g * \sin\left(\frac{pi * i}{18}\right);$$

Кінець

$$Npole_zovn = 73;$$

Для $i=0$ до 72 *Робити*

Початок

$$Xpole_zovn[i] = A_pole * \cos\left(\frac{pi * i}{36}\right);$$

$$Ypole_zovn[i] = A_pole * \sin\left(\frac{pi * i}{36}\right);$$

Кінець

Кінець

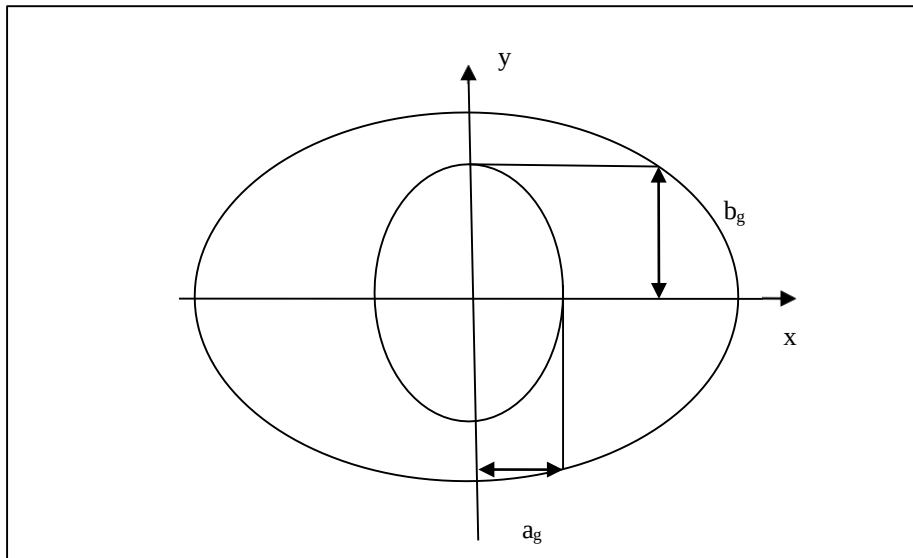


Рис 2.15. Поле ковбойської шляпи

Для побудови верха шляпи необхідно запроектувати викрійку, показану на рис. 2.16.

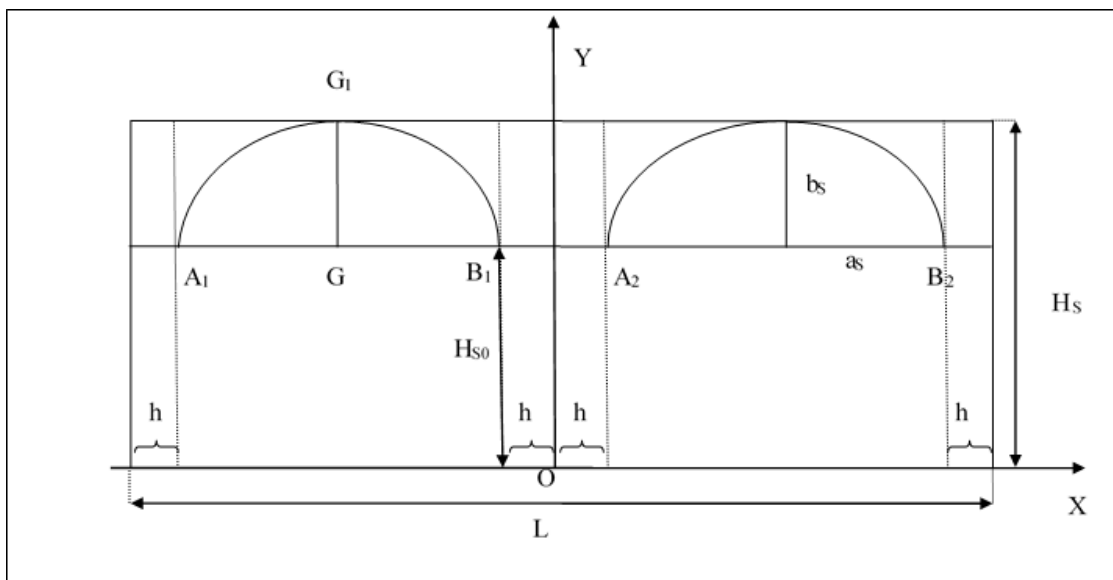


Рис. 2.16. Верх ковбойської шляпи

Будуємо прямокутник, довжина якого L , а висота - H_s . – вказується програмно. Ділимо його на дві рівні частини. Від осі OY та від бокових сторін прямокутника відкладаємо довжину h , як показано на рис. 2.16. Розраховуємо висоту H_{s0} для цього необхідно знайти b_s :

$$a_s = \frac{ObhGol - 2\Delta}{4}, \text{ де } \Delta = 10 - \text{припуски на шов; } b_s = H_s - H_{s0},$$

$$H_{s0} = H_s \cdot k_s, \quad k_s = 0,5. \text{ В свою чергу } L = ObhGol + 2\Delta.$$

Відрізок A_1B_1 та A_2B_2 ділимо пополам, і відкладаємо від нього перпендикуляр GG_1 довжиною b_s . Від точки G будуємо пів еліпс, з довжиною півосей a_s і b_s . Після побудови еліпсів, сполучаємо наші фігури. Програмно це реалізовано за допомогою наступного алгоритму[28]:

Початок

$$k_s = 0.5;$$

$$Hs0 = k_s * Hs;$$

$$A_s = \frac{ObhGol - 20}{4};$$

$$B_s = Hs - Hs0;$$

$$Xc1 = A_s + 10;$$

$$Yc1 = Hs0;$$

$$Xc2 = -A_s - 10;$$

$$Yc2 = Hs0;$$

Для $i=0$ до 36 Робити

Початок

$$Xverh_sh[i] = A_s * \cos\left(\frac{\pi * i}{36}\right) + Xc1;$$

$$Yverh_sh[i] = A_s * \sin\left(\frac{\pi * i}{36}\right) + Yc1;$$

Кінець

Для $i=0$ до 36 Робити

Початок

$$Xverh_sh[i + 37] = A_s * \cos\left(\frac{\pi * i}{36}\right) + Xc2;$$

$$Yverh_sh[i + 37] = A_s * \sin\left(\frac{\pi * i}{36}\right) + Yc2;$$

Кінець

$$X_{verh_sh}[74] = -2 * A_s - 20;$$

$$Y_{verh_sh}[74] = Hs0;$$

$$X_{verh_sh}[75] = -2 * A_s - 20;$$

$$Y_{verh_sh}[75] = 0;$$

$$X_{verh_sh}[76] = 2 * A_s + 20;$$

$$Y_{verh_sh}[76] = 0;$$

$$X_{verh_sh}[77] = 2 * A_s + 20;$$

$$Y_{verh_sh}[77] = Hs0;$$

$$X_{verh_sh}[78] = X_{verh_sh}[0];$$

$$Y_{verh_sh}[78] = Y_{verh_sh}[0];$$

$$N_{verh_sh} = 79;$$

Кінець

Для побудови дінця використаємо еліпс, а i і b якого розраховуються за формулами:

$$b = \sqrt{a_s b_s + \frac{20}{\pi}}; \quad a = b \cdot k.$$

Програмно це реалізовано за допомогою процедури наступного алгоритму:

Початок

$$B_nove = \sqrt{A_s * B_s + \frac{20}{pi}};$$

$$A_nove = k * B_nove;$$

Для i=0 до 72 Робити

$$X_dince[i] = A_nove * \cos(pi * \frac{i}{36});$$

Початок

Кінець Кінець.

$$Y_dince[i] = B_nove * \sin(pi * \frac{i}{36});$$

Панама

Для пошиття жіночої панамы необхідні такі складові: дві бокові стінки, окіл та верх панамы.

Для побудови бокової стінки побудуємо сектор з кутом 90 градусів, та проводимо розрахунки для визначення висоти сектора відносно обхвату голови(рис.2.17)[28].

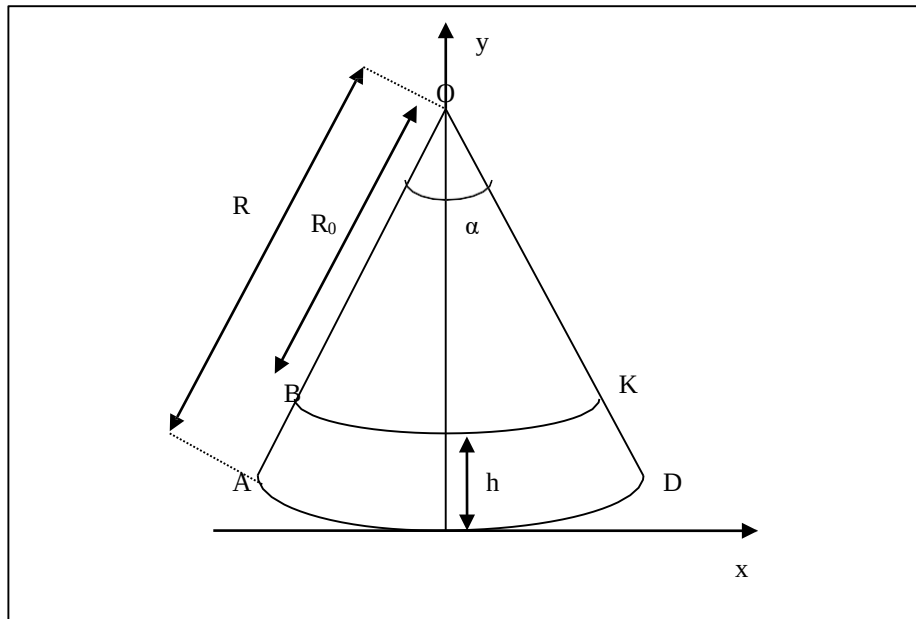


Рис. 2.17. Розгортка еліпса

Дуга $\cup AD = R \cdot \alpha$, тоді $\cup BK = R_0 \cdot \alpha$; $R = R_0 + h$. Отже

$$R_0 = \frac{h}{\frac{\pi \cdot R}{ObhGol} - 1}; \text{ де } h=90.$$

В результаті після побудови отримаємо окіл(рис.2.18):

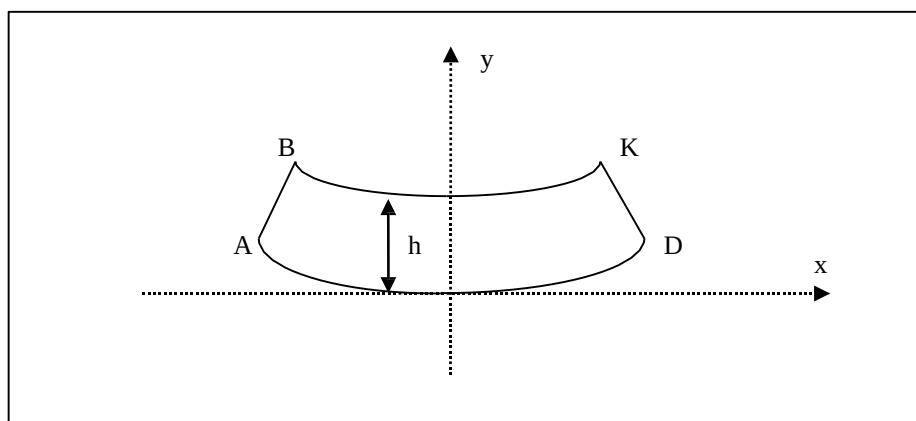


Рис. 2.18. Окіл панами

Програмна реалізація цього алгоритму наступна:

Початок

$$R_din = 137;$$

$$h = 90;$$

$$R0 = \frac{h}{\pi * R_din / ObhGol - 1};$$

$$R_rob = R0 + h;$$

$$Alfa = \pi / 2;$$

$$\varphi = Alfa / 18;$$

Для $i=0$ до 18 Робити

Початок

$$X_bortuk[i] = R_rob * \cos\left(\frac{\pi}{2} - \frac{Alfa}{2} + \varphi * i\right);$$

$$Y_bortuk[i] = R_rob * \sin\left(\frac{\pi}{2} - \frac{Alfa}{2} + \varphi * i\right);$$

Кінець

Для $i=0$ до 18 Робити

Початок

$$X_bortuk[19+i] = R0 * \cos\left(\pi - \frac{Alfa}{2} - \varphi * i\right);$$

$$Y_bortuk[19+i] = R0 * \sin\left(\pi - \frac{Alfa}{2} - \varphi * i\right);$$

Кінець

$$X_bortuk[38] = X_bortuk[0];$$

$$Y_bortuk[38] = Y_bortuk[0];$$

$$N_bortuka = 39;$$

Кінець

$$де R_din = R = 137;$$

Для побудови верху панамі використаємо уже знайомий алгоритм, і побудуємо коло радіусом $R = 137$ (рис.2.19).

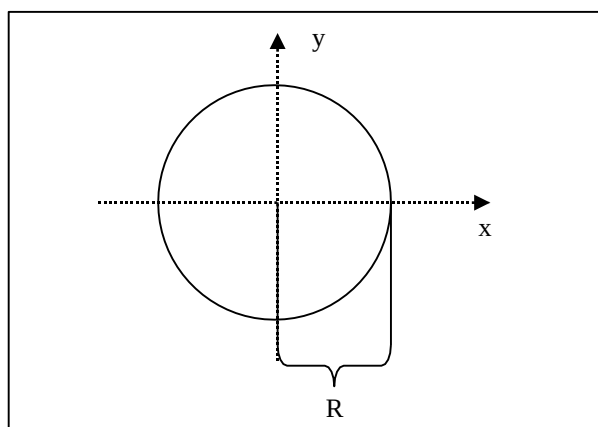


Рис. 2.19. Верхня частина панам (дінце)

Алгоритм для побудови дінця:

Початок

Для i=0 до 72 Робити

Початок

$X_dince[i] = r * \cos(pi * i / 36);$

Кінець Кінець

$Y_dince[i] = r * \sin(pi * i / 36);$

Тепер побудуємо околицю панам, як і в попередніх шляхах. Його ширина становитиме обхват голов из припуском на шов, а висота 50мм.

Шляпа-циліндр

Циліндр складається з трьох деталей – верху, бічної спинки та поля. Для їх побудови використаємо вже описані вище методи.

Побудова бічної стінки, здійснюється як показано на рис.2.20, де $H=250$, $\Delta=20$ мм.

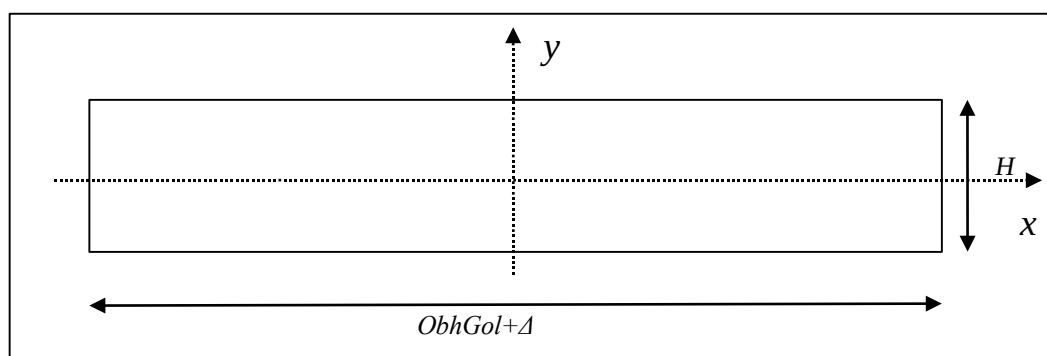


Рис 2.20. Бічна стінка циліндра

Поле циліндра, див. рис.2.21:

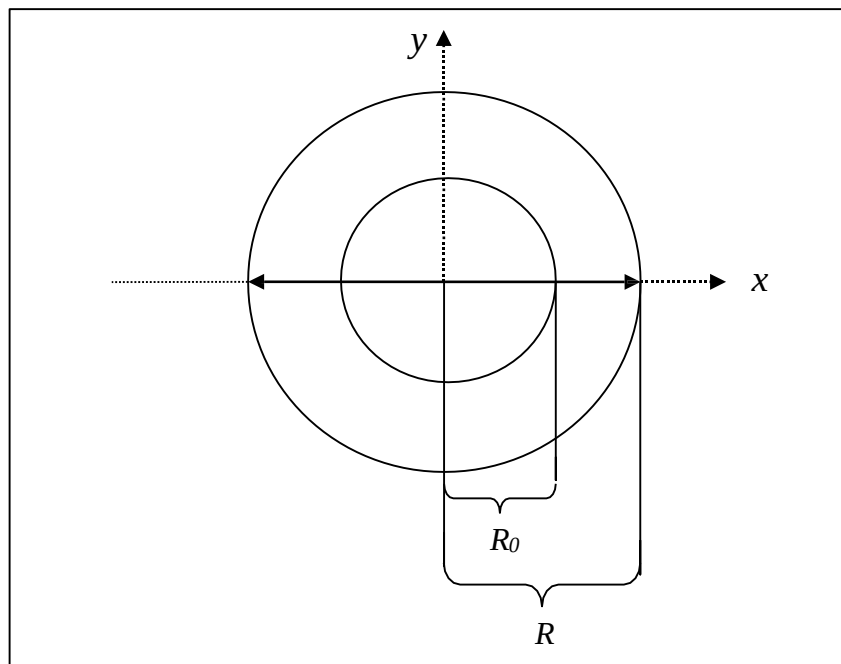


Рис. 2.21. Поле шляпи – циліндра

$$\text{де } R_0 = \frac{ObhGol}{2 \cdot \pi} + \Delta, R = R_0 + h, \text{ де } h \text{ – вводиться програмно.}$$

2.3. Задачі, що виникли при розробці алгоритмів для автоматизованого проєктування жіночих головних уборів[25-28]

Побудова відрізка заданої величини на прямій, перпендикулярній заданій прямій, що проходить через задану точку

При розробці алгоритмів для автоматизованого проєктування головних уборів виникли деякі задачі, які необхідно було вирішити. Розглянемо одну із них. Це побудова відрізка заданої величини на прямій, перпендикулярній заданій прямій, що проходить через задану точку

Дано: Пряма AB , що задана двома точками: $A(X_A; Y_A)$, $B(X_B; Y_B)$.

На прямій AB задана точка $C(X_C; Y_C)$.

З точки C провести відрізок, перпендикулярний прямій AB довжиною Δ (рис. 2.22).

Знайти: координати точки $D(X_D; Y_D)$.

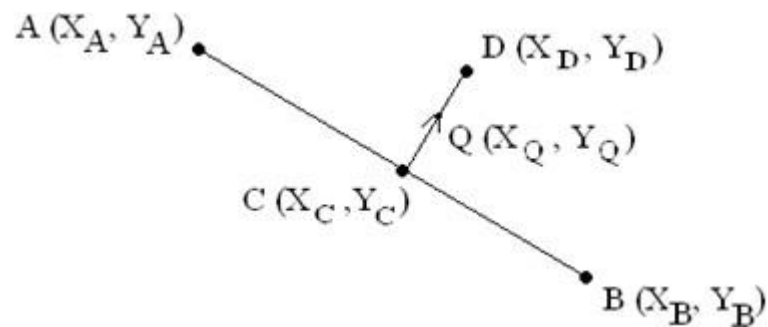


Рис. 2.22. Побудова відрізка заданої величини на прямій

Рівняння прямої AB , що проходить через дві точки можна представити наступним чином:

$$\frac{X - X_A}{X_B - X_A} = \frac{Y - Y_A}{Y_B - Y_A} \Rightarrow Ax + By + C = 0;$$

$$A = Y_B - Y_A;$$

$$B = X_A - X_B;$$

$$C = X_B \cdot Y_A - X_A \cdot Y_B.$$

Можливі три випадки:

а) Пряма AB паралельна вісі OX (рис. 2.23).

$$A = 0; By + C = 0 \Rightarrow Y = -\frac{C}{B}; Y_A = Y_B = Y_C = -\frac{C}{B}.$$

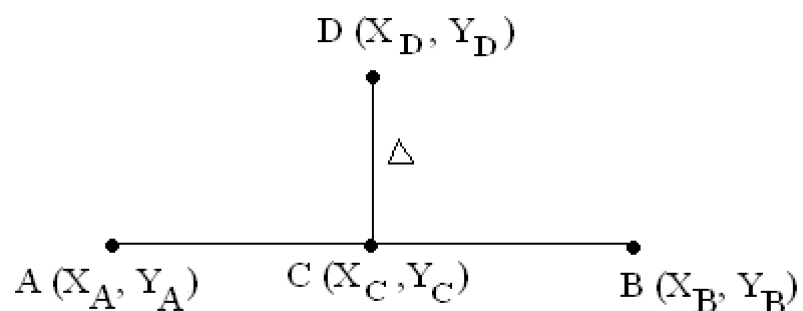


Рис. 2.23. Пряма AB паралельна вісі OX

Тоді $X_D = X_C$.

$$Y_D = Y_A \pm \Delta = -\frac{C}{B} \pm \Delta.$$

б) Пряма AB паралельна вісі OY (рис. 2.24).

$$B=0; Ax+C=0 \Rightarrow X = -\frac{C}{A}; \quad X_A = X_B = X_C = -\frac{C}{A};$$

$$X_D = X_C \pm \Delta = -\frac{C}{A} \pm \Delta; \quad Y_D = Y_C.$$

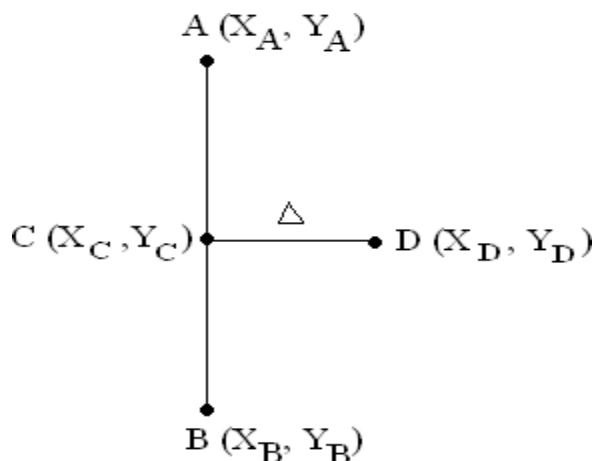


Рис. 2.24. Пряма AB паралельна вісі OY

в) Загальний випадок (пряма AB не паралельна осям) (рис.2.22).

Тоді рівняння прямої $CD \perp AB$ можна представити наступним чином:

$$CD: Y - Y_C = K_1 \cdot (X - X_C); \text{ де } K_1 \cdot K = -1 \Rightarrow K_1 = -\frac{1}{K}. \text{ Визначимо } K:$$

$$Ax + By + C = 0 \Rightarrow By = -Ax - C \Rightarrow Y = \frac{-Ax - C}{B} = -\underbrace{\frac{A}{B}}_K x - \frac{C}{B} = Kx - \frac{C}{B}$$

$$K = -\frac{A}{B};$$

Рівняння прямої CD має наступний вигляд: $Y - Y_C = \frac{X - X_C}{K}$ або

$$Y = Y_C + \frac{X - X_C}{K};$$

Визначимо довільну точку Q на прямій CD . Нехай: $X_Q = X_C \pm \Delta q$;

$$\text{Тоді: } Y_Q = Y_C + \frac{X_Q - X_C}{K} = Y_C \pm \frac{\Delta q}{K},$$

Тоді координати вектору $\overrightarrow{CQ}$ визватимуться наступним чином:

$$\overrightarrow{CQ} = \{X_Q - X_C; Y_Q - Y_C\};$$

Визначимо одиничний вектор на прямій CD : $\vec{e}_{CQ} = \left\{ \frac{X_Q - X_C}{D}; \frac{Y_Q - Y_C}{D} \right\}$,

$$\text{де } D = |\overline{CQ}| = \sqrt{(X_Q - X_C)^2 + (Y_Q - Y_C)^2}.$$

$$\overline{CD} = \{X_D - X_C; Y_D - Y_C\} = \Delta \cdot \vec{e}_{CQ} = \left\{ \frac{\Delta}{D} (X_Q - X_C); \frac{\Delta}{D} (Y_Q - Y_C) \right\};$$

$$\text{Звідси: } \begin{cases} X_D - X_C = \frac{\Delta}{D} (X_Q - X_C); \\ Y_D - Y_C = \frac{\Delta}{D} (Y_Q - Y_C) \end{cases} \Rightarrow \begin{cases} X_D = X_C + \frac{\Delta}{D} (X_Q - X_C); \\ Y_D = Y_C + \frac{\Delta}{D} (Y_Q - Y_C) \end{cases}$$

Висновки до другого розділу

1. Запропоновано алгоритми автоматизованого проектування головних уборів, а саме: беретів, панам (звичайна панама, панама із широкими полями та молодіжна панама), конфедератки шляпи-циліндр та ковбойської шляпи.
2. Розроблено алгоритми допоміжних задач, які дозволили успішно реалізувати алгоритми автоматизованого проектування головних уборів.

РОЗДІЛ 3. ПРОГРАМНЕ ЗАБЕЗПЕЧЕННЯ ДЛЯ АВТОМАТИЗОВАНОГО ПРОЄКТУВАННЯ ГОЛОВНИХ УБОРІВ

3.1. Програмні засоби

Використавши запропоновані алгоритми автоматизованого проєктування головних уборів за індивідуальними розмірними ознаками голови людини, для якої проєктується головний із асортименту запропонованих головних уборів, був розроблений програмний продукт для автоматизованого проєктування головних уборів. Для реалізації даного програмного продукту необхідні наступні програмні засоби: Delphi 7.0.

Цей вибір обґрунтовано тим, що використання даних інструментальних засобів дозволяє створити інформаційну систему для обробки і аналізу необхідного взаємопов'язаного набору даних[29].

Як операційна система була обрана ітерактивна оболонка Microsoft Windows – 2010, що дозволяє підтримувати режим багатозадачності, а також на відзнаку від відомих операційних систем Unix, OS/2 та ін. Windows забезпечує високу якість зображення. Вона володіє розширеними можливостями при паралельній роботі декількох програм, спрощеною передачею результатів від програми до програми, підвищеною ємністю системи, дозволяє водночас виконувати велике число прикладних програм і системних компонентів, ефективними засобами управління оперативною пам'яттю і іншими можливостями, що відрізняють її від вищезгаданих операційних систем.

Delphi 7.0 – це програмний інструмент для створення прикладних програм, які працюють в середовищі операційної системи Microsoft Windows – 2010. Пакет включає не тільки мову програмування, а й ефективне діалогове середовище для розробки форм та вікон, що дозволяє програмісту створити програмний продукт достатньо високої якості за порівняно невеликий проміжок часу. Delphi 7.0 дає можливість використовувати ресурси інших прикладних програм Windows та взаємодіяти з зовнішньою базою даних.

3.2. Структурна схема програмного продукту

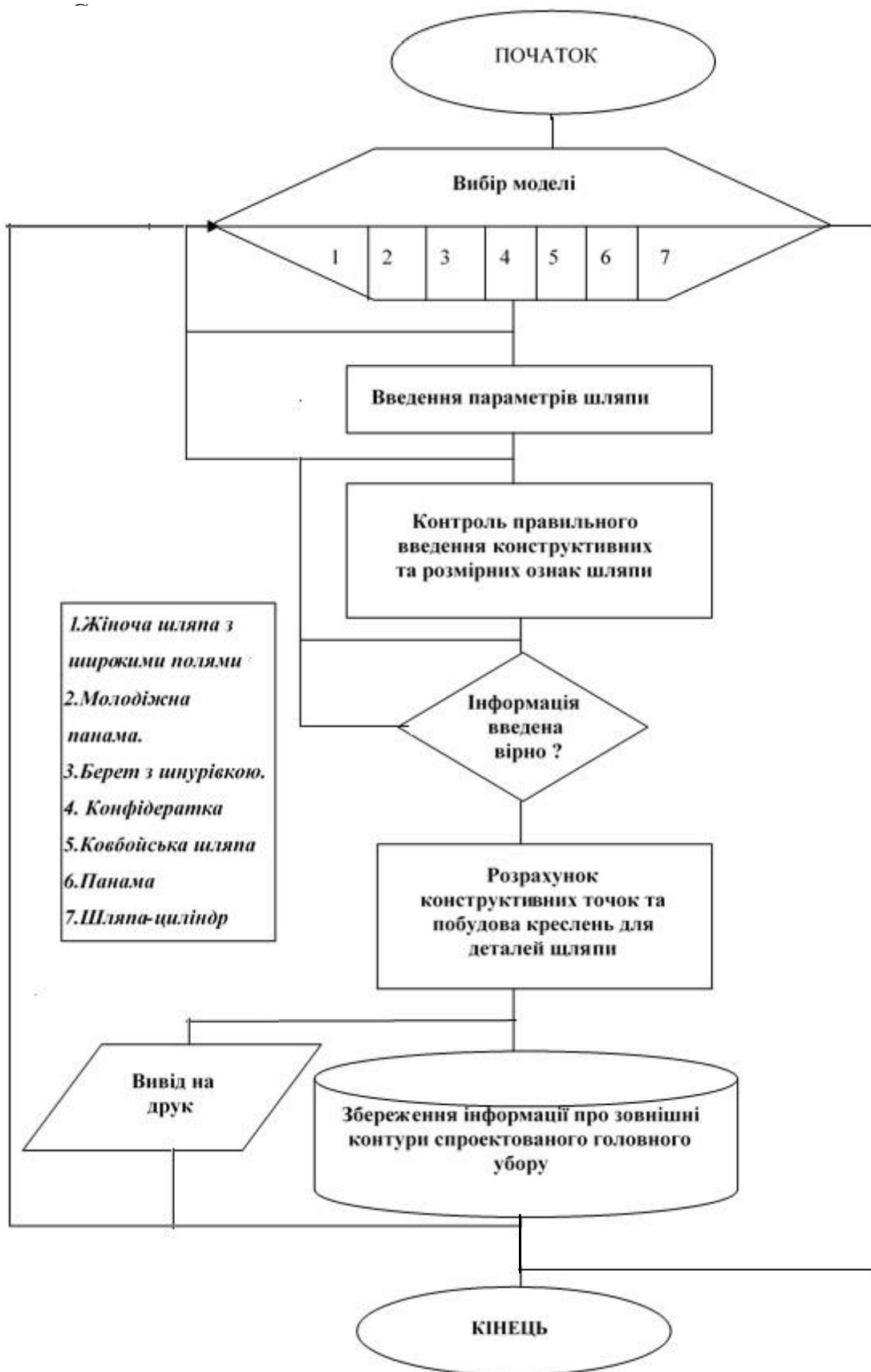


Рис. 3.1. Структурна схема програмного продукту

3.3. Основні функції програмного продукту[30-33]

Процедура Procedure SmeshchenieVector($X_a, Y_a, X_b, Y_b, delta:real; var X_c, Y_c:real$) забезпечує пошук координат точки $C(X_c, Y_c)$ на прямій AB , яка зміщена від точки $B(X_b, Y_b)$ на величину $delta$. Сама процедура представлена нижче:

```

Procedure SmeshchenieVector( $X_a, Y_a, X_b, Y_b, delta:real; var X_c, Y_c:real$ );
var  $Dx, Dy, D, Edx, Edy:real$ ;
begin
   $Dx := X_b - X_a$ ;  $Dy := Y_b - Y_a$ ;
   $D := \text{sqrt}(\text{sqr}(Dx) + \text{sqr}(Dy))$ ;
   $Edx := Dx/D$ ;
   $Eddy := Dy/D$ ;
   $X_c := X_b + delta * Edx$ ;
   $Y_c := Y_b + delta * Edy$ ;
end;
```

Процедура Procedure CrossTwoLine($X_1, Y_1, X_2, Y_2, X_3, Y_3, X_4, Y_4:real; var X, Y:real$) забезпечує пошук координат точки перетину двох прямих (X, Y) : прямої з координатами вершин (X_1, Y_1) та (X_2, Y_2) і прямою з координатами вершин (X_3, Y_3) та (X_4, Y_4) . Сама процедура представлена нижче:

```

Procedure CrossTwoLine( $X_1, Y_1, X_2, Y_2, X_3, Y_3, X_4, Y_4:real; var X, Y:real$ );
var  $delta1, delta2, delta, A1, A2, B1, B2, C1, C2:real$ ;
begin
   $A1 := y_1 - y_2$ ;  $A2 := y_3 - y_4$ ;
   $B1 := x_2 - x_1$ ;  $B2 := x_4 - x_3$ ;
   $C1 := x_1 * y_2 - x_2 * y_1$ ;  $C2 := x_3 * y_4 - x_4 * y_3$ ;
   $delta := A1 * B2 - A2 * B1$ ;
   $delta1 := B1 * C2 - B2 * C1$ ;
   $delta2 := A1 * C2 - A2 * C1$ ;
   $X := delta1 / delta$ ;
   $Y := delta2 / delta$ ;
end;
```

Процедура Procedure Perpend($X_a, Y_a, X_b, Y_b, delta:real; var Xc1, Yc1, Xc2, Yc2:real$) забезпечує пошук координат точок $(Xc1, Yc1)$, $(Xc2, Yc2)$ на прямій, що перпендикулярна прямій AB та які віддалені від неї на величину $delta$, яка задається координатами вершин $A(X_a, Y_a)$ та $B(X_b, Y_b)$. Сама процедура представлена нижче:

```

Procedure Perpend( $X_a, Y_a, X_b, Y_b, delta:real; var Xc1, Yc1, Xc2, Yc2:real$ );
Var  $k, Xd1, Yd1, Xd2, Yd2, Dbd:real$ ;
```

```

begin
if Xa=Xb then
begin
Xc1:=Xb+delta;
Xc2:=Xb-delta;
Yc1:=Yb;
Yc2:=Yb;
end
else
if Ya=Yb then
begin
Yc1:=Yb+delta;
Yc2:=Yb-delta;
Xc1:=Xb;
Xc2:=Xb;
end
else
begin
k:=(Yb-Ya)/(Xb-Xa);
Xd1:=Xb+5;
Xd2:=Xb-5;
Yd1:=Yb-(Xd1-Xb)/k;
Yd2:=Yb-(Xd2-Xb)/k;
SmeshchenieVector(Xd1,Yd1,Xb,Yb,delta,Xc1,Yc1);
SmeshchenieVector(Xd2,Yd2,Xb,Yb,delta,Xc2,Yc2);
end;
end;

```

Процедура Procedure Graf_im1(n:integer; x,y: array of real; MXY,Xcf,Ycf, XcE,YcE:real;P,Q,T:integer) забезпечує вивід в область Form1.Image1 многокутника з кількістю вершин n та координатами вершин, що знаходяться в масивах x,y.

Інші параметри процедури:

MXY- масштаб, в якому буде виводиться заданий многокутник в область Form1.Image1;

Xcf,Ycf - координати центру прямокутника, що описаний навколо заданого многокутника;

XcE,YcE - координати центру області Form1.Image1;

P – визначає колір сторін многокутника при виводі його креслення в область Form1.Image1;

Q – визначає колір вершин багатокутника при виводі його креслення в область *Form1.Image1*;

T – визначає товщину ліній сторін при виводі його креслення в область *Form1.Image1*.

Сама процедура представлена нижче:

```

Procedure Graf_im1(n:integer;x,y: array of real;
  MXY,Xcf,Ycf,XcE,YcE:real;P,Q,T:integer);
  var
    i:integer;
  X_r,Y_r: array[0..100] of integer;
  begin
    for i:=0 to n-1 do
      begin
        X_r[i]:=Round((x[i]-Xcf)*MXY+XcE);
        Y_r[i]:=Round((-Y[i]+Ycf)*MXY+YcE);
      end;
    for i:=0 to n-1 do
      with Form1.Image1.Canvas do
        begin
          if T=1 Then
            Pen.Width:=1
          else Pen.Width:=3;
          if P=1 Then
            Pen.Color:=clBlack
          else
            Pen.Color:=clBlue;
          if i=0 then
            MoveTo(X_r[i],Y_r[i])
          else LineTo(X_r[i],Y_r[i])
          end;
          if Q=1 Then
            for i:=0 to n-1 do
              with Form1.Image.Canvas do
                begin
                  Pen.Color:=ClRed;
                  Ellipse(X_r[i]-2,Y_r[i]-2,X_r[i]+2,Y_r[i]+2);
                end;
              end;
            end;
          end;
        end;
      end;
    end;
  end;

```

Процедура *procedure TForm1.Button3Click(Sender: TObject)* забезпечує збереження у файлі інформацію про зовнішні контури деталей моделі.

```

procedure TForm1.Button3Click(Sender: TObject);
    Var F:system.Text;
    Name:string[100];
    i,j:integer;
    begin
        If SaveDialog1.Execute then
            begin
                Name:=SaveDialog1.FileName;
                System.Assign(f,name);
                Rewrite(f);
                Writeln(f,model);
                Writeln(f);
                Writeln(f,col_det);
                for i:=1 to col_det do
                    Writeln(f,name_det[i]);
                for i:=1 to col_det do
                    Writeln(f,col_point_det[i]);
                for i:=1 to col_det do
                    for j:=0 to col_point_det[i]-1 do
                        if j=0 then
                            Writeln(f,x_det[i,j]:6:2,'    ',y_det[i,j]:6:2,'    ',name_det[i])
                        else
                            Writeln(f,x_det[i,j]:6:2,'    ',y_det[i,j]:6:2);
                    System.Close(f);
                end;
            end;
    end;

```

Нижче представлені процедури та функції, які забезпечують вивід креслень деталей зпроектованих моделей головних уборів

```

function MashPrint(dl_f,sh_f,dl_E,sh_E:real):real;
    var MX,MY:real;
    begin
        MX:=dl_E/dl_f;
        MY:=sh_E/sh_f;
        if MX<MY then MashPrint:=MX
        else MashPrint:=MY;
    end;

```

```

Procedure PrintKonfideratka;
    var i:integer;
    Xr,Yr:array[0..200] of integer;
    MXY_PTR:real;
    begin
        GabRozmiru;

```

```

M_XY[1]:=MashPrint(Dovg[1],Vusot[1],Printer.PageWidth/2,Printer.PageHeight/2
);
M_XY[2]:=MashPrint(Dovg[2],Vusot[2],Printer.PageWidth/2,Printer.PageHeight/2
);
    for i:=1 to 2 do
    if i=1 then MXY_PTR:=M_XY[1]
    else
    if MXY_PTR>M_XY[i] then
    MXY_PTR:=M_XY[i];
    MXY_PTR:=MXY_PTR;
    With Printer.Canvas Do
    begin
    Font.Name:='Arial';
    Font.Size:=20;
    Font.Color:=clBlack;
    TextOut(Round(1/50*Printer.PageWidth),
    Round(1/50*Printer.PageHeight),'КОНФІДЕРАТКА');
    Font.Name:='Arial';
    Font.Size:=16;
    Font.Color:=clBlack;
    for i:=0 to col_point_det[1]-1 do
    begin
    Xr[i]:=round((x_det[1,i]-Xcf[1])*MXY_PTR + Printer.PageWidth/2);
    Yr[i]:=round((y_det[1,i]-Ycf[1])*MXY_PTR + Printer.PageHeight/3-300);
    end;
    for i:=0 to col_point_det[1]-1 do
    if i=0 then MoveTo(Xr[i],Yr[i])
    else LineTo(Xr[i],Yr[i]);
    TextOut(Round(1/2*Printer.PageWidth)-400, Round(1/2*Printer.PageHeight)-
    100,'1.Верхня частина конфідератки');
    for i:=0 to col_point_det[2]-1 do
    begin
    Xr[i]:=round((x_det[2,i]-Xcf[2])*MXY_PTR + Printer.PageWidth/2);
    Yr[i]:=round((y_det[2,i]-Ycf[2])*MXY_PTR + 4/5*Printer.PageHeight);
    end;
    for i:=0 to col_point_det[2]-1 do
    if i=0 then MoveTo(Xr[i],Yr[i])
    else LineTo(Xr[i],Yr[i]);
    TextOut(Round(1/2*Printer.PageWidth)-400,
    Round(5/6*Printer.PageHeight)+300,'2.Бокова сторона конфідератки');
    end;
    end;

```

//-----

```

Procedure PrintShlapaNeshurPole;
    var i:integer;
    Xr,Yr:array[0..200] of integer;
    MXY_PTR:real;
    begin
        GabRozmiru;
M_XY[1]:=MashPrint(Dovg[1],Vusot[1],Printer.PageWidth/2,Printer.PageHeight/2
    );
M_XY[2]:=MashPrint(Dovg[2],Vusot[2],Printer.PageWidth/2,Printer.PageHeight/2
    );
M_XY[3]:=MashPrint(Dovg[3],Vusot[3],Printer.PageWidth/2,Printer.PageHeight/2
    );
M_XY[4]:=MashPrint(Dovg[4],Vusot[4],Printer.PageWidth/2,Printer.PageHeight/2
    );
        for i:=1 to 4 do
            if i=1 then MXY_PTR:=M_XY[1]
                else
                    if MXY_PTR>M_XY[i] then
                        MXY_PTR:=M_XY[i];
                    MXY_PTR:=MXY_PTR;
                    With Printer.Canvas Do
                        begin
                            Font.Name:='Arial';
                            Font.Size:=20;
                            Font.Color:=clBlack;
                            TextOut(Round(1/50*Printer.PageWidth),
                                Round(1/50*Printer.PageHeight),'ЖИНОЧА ПАНАМА З НЕШИИРОКИМИ
                                ПОЛЯМИ' );
                            Font.Name:='Arial';
                            Font.Size:=16;
                            Font.Color:=clBlack;
                            for i:=0 to col_point_det[1]-1 do
                                begin
                                    Xr[i]:=round((x_det[1,i]-Xcf[1])*MXY_PTR + 1/4*Printer.PageWidth);
                                    Yr[i]:=round((y_det[1,i]-Ycf[1])*MXY_PTR + 4/5*Printer.PageHeight);
                                end;
                                for i:=0 to col_point_det[1]-1 do
                                    if i=0 then MoveTo(Xr[i],Yr[i])
                                        else LineTo(Xr[i],Yr[i]);
                                    TextOut(Round(1/4*Printer.PageWidth)-100,
                                        Round(4/5*Printer.PageHeight)+400,'3.Клин(1)');
                                    for i:=0 to col_point_det[2]-1 do
                                        begin
                                            Xr[i]:=round((x_det[2,i]-Xcf[2])*MXY_PTR + 3/4*Printer.PageWidth);
                                            Yr[i]:=round((y_det[2,i]-Ycf[2])*MXY_PTR + 4/5*Printer.PageHeight);

```



```

        end;
        for i:=0 to col_point_det[2]-1 do
            if i=0 then MoveTo(Xr[i],Yr[i])
            else LineTo(Xr[i],Yr[i]);
            TextOut(Round(3/4*Printer.PageWidth)-200,
                Round(4/5*Printer.PageHeight)+400,'4.Клин(2)');
            for i:=0 to col_point_det[3]-1 do
                begin
                    Xr[i]:=round((x_det[3,i]-Xcf[3])*MXY_PTR + Printer.PageWidth/2);
                    Yr[i]:=round((y_det[3,i]-Ycf[3])*MXY_PTR + 1/6*Printer.PageHeight);
                end;
                for i:=0 to col_point_det[3]-1 do
                    if i=0 then MoveTo(Xr[i],Yr[i])
                    else LineTo(Xr[i],Yr[i]);
                    TextOut(Round(Printer.PageWidth/2)-300,
                        Round(1/6*Printer.PageHeight)+250,'1.Основа панаму');
                    for i:=0 to col_point_det[4]-1 do
                        begin
                            Xr[i]:=round((x_det[4,i]-Xcf[4])*MXY_PTR + 1/2*Printer.PageWidth);
                            Yr[i]:=round((y_det[4,i]-Ycf[4])*MXY_PTR + 2/3*Printer.PageHeight);
                        end;
                        for i:=0 to col_point_det[4]-1 do
                            if i=0 then MoveTo(Xr[i],Yr[i])
                            else LineTo(Xr[i],Yr[i]);
                            TextOut(Round(1/2*Printer.PageWidth)-300,
                                Round(2/3*Printer.PageHeight),'2.Поле панаму');
                        end;
                    end;
                end;
            Procedure PrintShlapaShurPole;
                var i:integer;
                Xr,Yr:array[0..200] of integer;
                MXY_PTR:real;
                begin
                    GabRozmiru;
                    M_XY[1]:=MashPrint(Dovg[1],Vusot[1],Printer.PageWidth/2,Printer.PageHeight/2);
                end;
                M_XY[2]:=MashPrint(Dovg[2],Vusot[2],Printer.PageWidth/2,Printer.PageHeight/2);
                end;
                for i:=1 to 2 do
                    if i=1 then MXY_PTR:=M_XY[1]
                    else
                        if MXY_PTR>M_XY[i] then
                            MXY_PTR:=M_XY[i];
                        MXY_PTR:=MXY_PTR;
                        With Printer.Canvas Do

```

```

begin
  Font.Name:='Arial';
  Font.Size:=20;
  Font.Color:=clBlack;
  TextOut(Round(1/50*Printer.PageWidth),
Round(1/50*Printer.PageHeight),'ЖИНОЧА ШЛЯПА З ШИРОКИМИ ПОЛЯМИ'
);
  Font.Name:='Arial';
  Font.Size:=16;
  Font.Color:=clBlack;
  for i:=0 to col_point_det[1]-1 do
    begin
      Xr[i]:=round((x_det[1,i]-Xcf[1])*MXY_PTR + 1/4*Printer.PageWidth);
      Yr[i]:=round((y_det[1,i]-Ycf[1])*MXY_PTR + Printer.PageHeight/2);
      end;
    for i:=0 to col_point_det[1]-1 do
      if i=0 then MoveTo(Xr[i],Yr[i])
      else LineTo(Xr[i],Yr[i]);
      TextOut(Round(1/4*Printer.PageWidth)-200,
Round(1/2*Printer.PageHeight)+500,'1.Клин шляпи' );
      for i:=0 to col_point_det[2]-1 do
        begin
          Xr[i]:=round((x_det[2,i]-Xcf[2])*MXY_PTR + 3/4*Printer.PageWidth);
          Yr[i]:=round((y_det[2,i]-Ycf[2])*MXY_PTR + Printer.PageHeight/2);
          end;
        for i:=0 to col_point_det[2]-1 do
          if i=0 then MoveTo(Xr[i],Yr[i])
          else LineTo(Xr[i],Yr[i]);
          TextOut(Round(4/6*Printer.PageWidth)+100,
Round(1/2*Printer.PageHeight)+500,'2.Поле шляпи' );
          end;
        end;

```

//-----

```

  Procedure PrintThselindr;
    var i:integer;
    Xr,Yr:array[0..200] of integer;
    MXY_PTR:real;
    begin
      GabRozmiru;
      M_XY[1]:=MashPrint(Dovg[1],Vusot[1],2/3*Printer.PageWidth-
20,2/3*Printer.PageHeight-20);
      M_XY[2]:=MashPrint(Dovg[2],Vusot[2],2/3*Printer.PageWidth-
20,2/3*Printer.PageHeight-20);

```

```

M_XY[3]:=MashPrint(Dovg[3],Vusot[3],1/3*Printer.PageWidth-20,
Printer.PageHeight-20);
M_XY[4]:=MashPrint(Dovg[4],Vusot[4],2/3*Printer.PageWidth-
20,1/3*Printer.PageHeight-20);
for i:=1 to 4 do
if i=1 then MXY_PTR:=M_XY[1]
else
if MXY_PTR>M_XY[i] then
MXY_PTR:=M_XY[i];
MXY_PTR:=MXY_PTR;
With Printer.Canvas Do
begin
Font.Name:='Arial';
Font.Size:=16;
Font.Color:=clBlack;
for i:=0 to col_point_det[1]-1 do
begin
Xr[i]:=round((x_det[1,i]-Xcf[1])*MXY_PTR + Printer.PageWidth/3);
Yr[i]:=round((y_det[1,i]-Ycf[1])*MXY_PTR + Printer.PageHeight/3);
end;
for i:=0 to col_point_det[1]-1 do
if i=0 then MoveTo(Xr[i],Yr[i])
else LineTo(Xr[i],Yr[i]);
TextOut(Round(3/4*Printer.PageWidth),
Round(3*Printer.PageHeight/4),'1.Поле циліндра');
for i:=0 to col_point_det[2]-1 do
begin
Xr[i]:=round((x_det[2,i]-Xcf[3])*MXY_PTR + 5/6*Printer.PageWidth);
Yr[i]:=round((y_det[2,i]-Ycf[3])*MXY_PTR + Printer.PageHeight/2);
end;
for i:=0 to col_point_det[2]-1 do
if i=0 then MoveTo(Xr[i],Yr[i])
else LineTo(Xr[i],Yr[i]);
for i:=0 to col_point_det[3]-1 do
begin
Xr[i]:=round((x_det[3,i]-Xcf[3])*MXY_PTR + 5/6*Printer.PageWidth);
Yr[i]:=round((y_det[3,i]-Ycf[3])*MXY_PTR + Printer.PageHeight/2);
end;
for i:=0 to col_point_det[3]-1 do
if i=0 then MoveTo(Xr[i],Yr[i])
else LineTo(Xr[i],Yr[i]);
TextOut(Round(Printer.PageWidth/4+150), Round(Printer.PageHeight/2-100),'2.
'+name_det[3]);
for i:=0 to col_point_det[4]-1 do
begin

```

```

        Xr[i]:=round((x_det[4,i]-Xcf[4])*MXY_PTR + Printer.PageWidth/3);
        Yr[i]:=round((y_det[4,i]-Ycf[4])*MXY_PTR + 2/3*Printer.PageHeight);
        end;
        for i:=0 to col_point_det[4]-1 do
            if i=0 then MoveTo(Xr[i],Yr[i])
            else LineTo(Xr[i],Yr[i]);
        TextOut(Round(Printer.PageWidth/4), Round(3*Printer.PageHeight/4),'3.
            '+name_det[4]);
        end;
    end;
end;
//-----
Procedure PrintShlapaPanama;
    var i:integer;
    Xr,Yr:array[0..200] of integer;
    MXY_PTR:real;
    begin
        GabRozmiru;
        M_XY[1]:=MashPrint(Dovg[1],Vusot[1],2/3*Printer.PageWidth-
            20,1/2*Printer.PageHeight-20);
        M_XY[2]:=MashPrint(Dovg[2],Vusot[2],2/3*Printer.PageWidth-
            20,1/2*Printer.PageHeight-20);
        M_XY[3]:=MashPrint(Dovg[3],Vusot[3],1/3*Printer.PageWidth-20,
            Printer.PageHeight-20);
        for i:=1 to 3 do
            if i=1 then MXY_PTR:=M_XY[1]
            else
                if MXY_PTR>M_XY[i] then
                    MXY_PTR:=M_XY[i];
                With Printer.Canvas Do
                    begin
                        Font.Name:='Arial';
                        Font.Size:=20;
                        Font.Color:=clBlack;
                        TextOut(Round(1/50*Printer.PageWidth),
                            Round(1/50*Printer.PageHeight),'ШЛЯПА - ПАНАМА');
                        Font.Name:='Arial';
                        Font.Size:=16;
                        Font.Color:=clBlack;
                    end;
                for i:=0 to col_point_det[1]-1 do
                    begin
                        Xr[i]:=round((x_det[1,i]-Xcf[1])*MXY_PTR + 1/3*Printer.PageWidth);
                        Yr[i]:=round((y_det[1,i]-Ycf[1])*MXY_PTR + 1/4*Printer.PageHeight);
                    end;
                for i:=0 to col_point_det[1]-1 do
                    if i=0 then MoveTo(Xr[i],Yr[i])

```

```

        else LineTo(Xr[i],Yr[i]);
        TextOut(Round(1/4*Printer.PageWidth)+50,
        Round(1/3*Printer.PageHeight),'1.Поле шляпи');
        for i:=0 to col_point_det[2]-1 do
            begin
                Xr[i]:=round((x_det[2,i]-Xcf[2])*MXY_PTR + 5/6*Printer.PageWidth);
                Yr[i]:=round((y_det[2,i]-Ycf[2])*MXY_PTR + Printer.PageHeight/2);
            end;
        for i:=0 to col_point_det[2]-1 do
            if i=0 then MoveTo(Xr[i],Yr[i])
            else LineTo(Xr[i],Yr[i]);
            TextOut(Round(3/4*Printer.PageWidth)+150,
            Round(2/3*Printer.PageHeight),'2.Дінець');
            for i:=0 to col_point_det[3]-1 do
                begin
                    Xr[i]:=round((x_det[3,i]-Xcf[3])*MXY_PTR + 1/3*Printer.PageWidth);
                    Yr[i]:=round((y_det[3,i]-Ycf[3])*MXY_PTR + 3/4*Printer.PageHeight);
                end;
            for i:=0 to col_point_det[3]-1 do
                if i=0 then MoveTo(Xr[i],Yr[i])
                else LineTo(Xr[i],Yr[i]);
                TextOut(Round(1/4*Printer.PageWidth)+50,
                Round(3/4*Printer.PageHeight)+200,'3.Окіл панамми');
            end;
        end;

//-----
        Procedure PrintKovbojska;
        var i:integer;
        Xr,Yr:array[0..200] of integer;
        MXY_PTR:real;
        begin
            GabRozmiru;
            M_XY[1]:=MashPrint(Dovg[1],Vusot[1],2/3*Printer.PageWidth-
            20,2/3*Printer.PageHeight-20);
            M_XY[2]:=MashPrint(Dovg[2],Vusot[2],2/3*Printer.PageWidth-
            20,2/3*Printer.PageHeight-20);
            M_XY[3]:=MashPrint(Dovg[3],Vusot[3],1/3*Printer.PageWidth-20,
            Printer.PageHeight-20);
            //M_XY[4]:=MashPrint(Dovg[4],Vusot[4],2/3*Printer.PageWidth-
            20,1/3*Printer.PageHeight-20);
            M_XY[4]:=MashPrint(Dovg[4],Vusot[4],Printer.PageWidth/2,Printer.PageHeight/2
            );
            for i:=1 to 4 do

```

```

if i=1 then MXY_PTR:=M_XY[1]
else
if MXY_PTR>M_XY[i] then
MXY_PTR:=M_XY[i];
MXY_PTR:=MXY_PTR;
With Printer.Canvas Do
begin
Font.Name:='Arial';
Font.Size:=20;
Font.Color:=clBlack;
TextOut(Round(1/50*Printer.PageWidth),
Round(1/50*Printer.PageHeight),'КОББОЙСЬКА ШЛЯПА');
Font.Name:='Arial';
Font.Size:=16;
Font.Color:=clBlack;
for i:=0 to col_point_det[1]-1 do
begin
Xr[i]:=round((x_det[1,i]-Xcf[1])*MXY_PTR + Printer.PageWidth*1/4);
Yr[i]:=round((y_det[1,i]-Ycf[1])*MXY_PTR + Printer.PageHeight/3+200);
end;
for i:=0 to col_point_det[1]-1 do
if i=0 then MoveTo(Xr[i],Yr[i])
else LineTo(Xr[i],Yr[i]);
for i:=0 to col_point_det[2]-1 do
begin
Xr[i]:=round((x_det[2,i]-Xcf[2])*MXY_PTR + Printer.PageWidth*1/4);
Yr[i]:=round((y_det[2,i]-Ycf[2])*MXY_PTR + Printer.PageHeight/3+200);
end;
for i:=0 to col_point_det[2]-1 do
if i=0 then MoveTo(Xr[i],Yr[i])
else LineTo(Xr[i],Yr[i]);
TextOut(Round(1/4*Printer.PageWidth)-250,
Round(Printer.PageHeight*2/3)+100,'1.Поле шляпи');
for i:=0 to col_point_det[3]-1 do
begin
Xr[i]:=round((x_det[3,i]-Xcf[3])*MXY_PTR + 5/6*Printer.PageWidth);
Yr[i]:=round((y_det[3,i]-Ycf[3])*MXY_PTR + 3/4*Printer.PageHeight);
end;
for i:=0 to col_point_det[3]-1 do
if i=0 then MoveTo(Xr[i],Yr[i])
else LineTo(Xr[i],Yr[i]);
TextOut(Round(5/6*Printer.PageWidth)-200,
Round(3/4*Printer.PageHeight)+200,'2. '+name_det[3]);
for i:=0 to col_point_det[4]-1 do
begin

```

```

Xr[i]:=round((x_det[4,i]-Xcf[4])*MXY_PTR + 5/6*Printer.PageWidth);
Yr[i]:=round((y_det[4,i]-Ycf[4])*MXY_PTR + 1/4*Printer.PageHeight);
    end;
    for i:=0 to col_point_det[4]-1 do
        if i=0 then MoveTo(Xr[i],Yr[i])
        else LineTo(Xr[i],Yr[i]);
        TextOut(Round(5/6*Printer.PageWidth)-100,
Round(1/4*Printer.PageHeight)+300,'3. '+name_det[4]);
    end;
end;

```

//-----

```

    Procedure PrintPanamaShnurovka;
        var i:integer;
        Xr,Yr:array[0..200] of integer;
        MXY_PTR:real;
        begin
            GabRozmiru;
M_XY[1]:=MashPrint(Dovg[1],Vusot[1],Printer.PageWidth/2,Printer.PageHeight/2
        );
M_XY[2]:=MashPrint(Dovg[2],Vusot[2],Printer.PageWidth/2,Printer.PageHeight/2
        );
M_XY[3]:=MashPrint(Dovg[3],Vusot[3],Printer.PageWidth/2,Printer.PageHeight/2
        );
        for i:=1 to 3 do
            if i=1 then MXY_PTR:=M_XY[1]
            else
                if MXY_PTR>M_XY[i] then
                    MXY_PTR:=M_XY[i];
                With Printer.Canvas Do
                    begin
                        Font.Name:='Arial';
                        Font.Size:=20;
                        Font.Color:=clBlack;
                        TextOut(Round(1/50*Printer.PageWidth),
Round(1/50*Printer.PageHeight),'ЖИНОЧА ПАНАМА З ДЕКОРАТИВНОЮ
ШИНУРОВКОЮ');
                        Font.Name:='Arial';
                        Font.Size:=16;
                        Font.Color:=clBlack;
                    for i:=0 to col_point_det[1]-1 do
                        begin
M_XY[i]:=round((x_det[1,i]-Xcf[1])*MXY_PTR + 1/4*Printer.PageWidth);
Yr[i]:=round((y_det[1,i]-Ycf[1])*MXY_PTR + 4/5*Printer.PageHeight);

```

```

end;
for i:=0 to col_point_det[1]-1 do
  if i=0 then MoveTo(Xr[i],Yr[i])
  else LineTo(Xr[i],Yr[i]);
  TextOut(Round(Printer.PageWidth/4)-200,
  Round(4/5*Printer.PageHeight)+350,'2.Клин панаму');
  for i:=0 to col_point_det[2]-1 do
    begin
      Xr[i]:=round((x_det[2,i]-Xcf[2])*MXY_PTR + 3/4*Printer.PageWidth);
      Yr[i]:=round((y_det[2,i]-Ycf[2])*MXY_PTR + 4/5*Printer.PageHeight);
    end;
    for i:=0 to col_point_det[2]-1 do
      if i=0 then MoveTo(Xr[i],Yr[i])
      else LineTo(Xr[i],Yr[i]);
      TextOut(Round(3/4*Printer.PageWidth)-400,
      Round(4/5*Printer.PageHeight)+350,'3.Клин панаму з шнуровкою');
      for i:=0 to col_point_det[3]-1 do
        begin
          Xr[i]:=round((x_det[3,i]-Xcf[3])*MXY_PTR + 1/2*Printer.PageWidth);
          Yr[i]:=round((y_det[3,i]-Ycf[3])*MXY_PTR + 1/3*Printer.PageHeight);
        end;
        for i:=0 to col_point_det[3]-1 do
          if i=0 then MoveTo(Xr[i],Yr[i])
          else LineTo(Xr[i],Yr[i]);
          TextOut(Round(1/2*Printer.PageWidth)-200,
          Round(1/3*Printer.PageHeight)+200,'1.Окіл панаму');
        end;
      end;
      begin
        {}
      if PrintDialog1.Execute then
        begin
          Printer.Orientation:=poLandscape;
          Printer.BeginDoc;
          if pr_Im11 then PrintThselindr
          else
          if pr_Im9 then PrintShlapaPanama
          else
          if Pr_Im8 then PrintKonfideratka
          else
          if Pr_Im3 then PrintShlapaNeshurPole
          else
          if Pr_Im2 then PrintShlapaShurPole
          else
          if Pr_Im6 then PrintPanamaShnurovka

```



```

else
if Pr_Im7 then PrintKovbojska;
//GrafPrt(k);
Printer.EndDoc;
end;
end;

```

3.4. Інструкція користувачу по роботі з програмним продуктом для автоматизованого проєктування головних уборів

На рис. 3.2 представлена основна форма програмного продукту для автоматизованого проєктування головних уборів.

Після вибору однієї із моделей головного убору, ми переходимо до закладки для введення основних розмірів головного убору (рис. 3.3).



Рис. 3.2. Основна форма програмного продукту

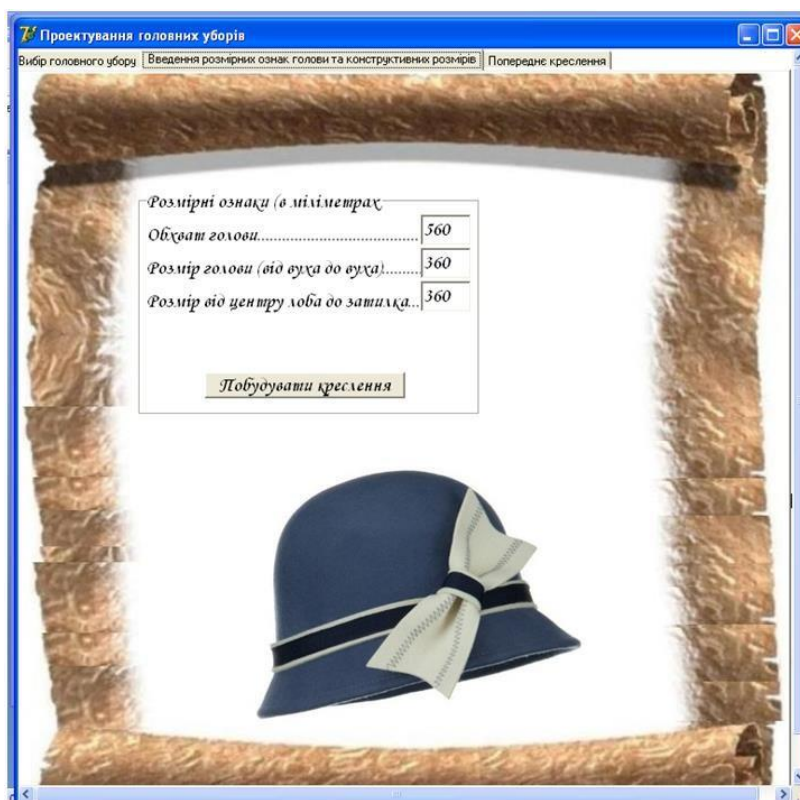


Рис. 3.3. Введення розмірних ознак голови

Після введення цих параметрів отримаємо креслення деталей головного убору(рис. 3.4)

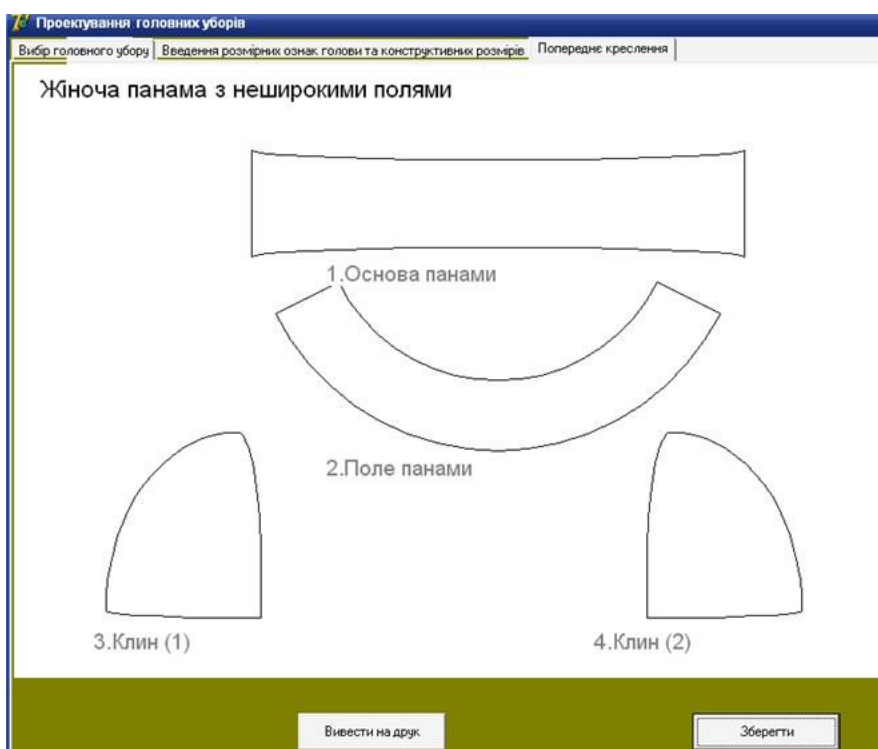


Рис. 3.4. Креслення деталей жіночої панамы з неширокими полями

Інформацію про деталі головного убору можна зберегти у файлі формату сумісному з системами автоматизованого проектування раціональних схем розкрою для виробів легкої промисловості (рис. 3.5) або вивести на друк.

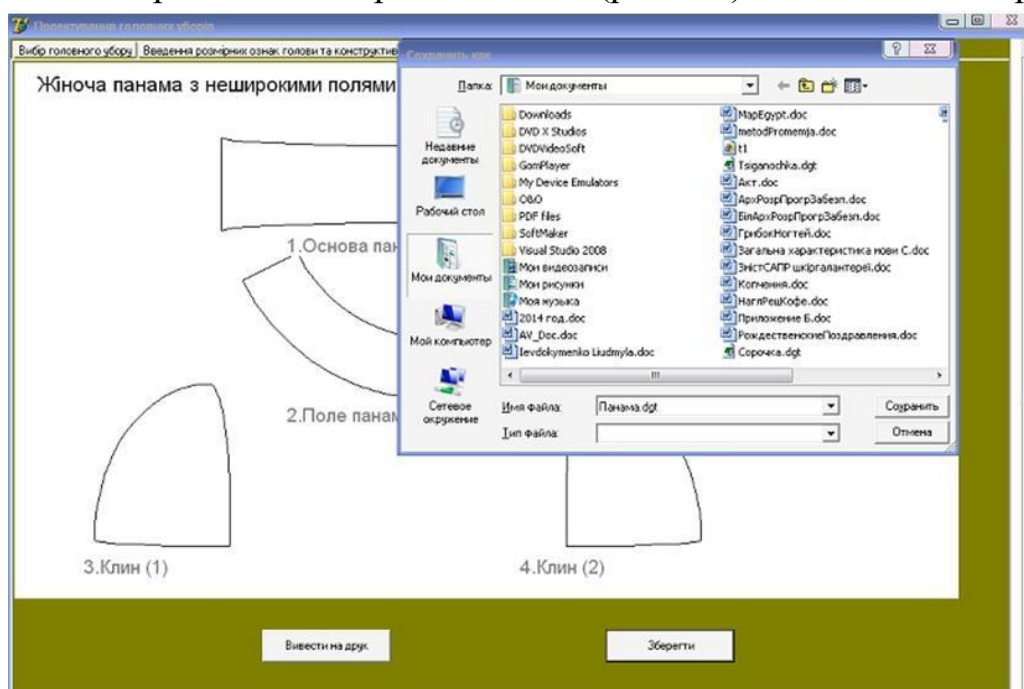


Рис. 3.5. Збереження у файлі інформації про зовнішні контури деталей жіночої панамки з неширокими полями

Збережена інформація про зовнішні контури деталей жіночої панамки має наступний вигляд:

Жіноча панамка з неширокими полями – назва моделі

4 – кількість деталей в моделі

Клин (1) – назва першої по порядку деталі в моделі

Клин (2) – назва другої по порядку деталі в моделі

Основа панамки – назва третьої по порядку деталі в моделі

Поле панамки – назва четвертої по порядку деталі в моделі

48 – кількість вершин на зовнішньому контурі в першій по порядку деталі в моделі

48 – кількість вершин на зовнішньому контурі в другій по порядку деталі в моделі

39 – кількість вершин на зовнішньому контурі в третій по порядку деталі в моделі

39 – кількість вершин на зовнішньому контурі в четвертій по порядку деталі в моделі

17.00 0.00 *Клин (1)* – координати X та Y вершин першої деталі

16.94 8.02

16.74 15.98

16.42 23.81

15.97 31.47

15.41 38.88

14.72 46.00

13.93 52.77

13.02	59.14
12.02	65.05
10.93	70.48
9.75	75.36
8.50	79.67
7.18	83.38
5.81	86.45
4.40	88.87
2.95	90.60
1.48	91.65
0.00	92.00
-8.11	91.52
-16.15	90.07
-24.07	87.67
-31.81	84.34
-39.30	80.10
-46.50	74.99
-53.34	69.03
-59.78	62.29
-65.76	54.80
-71.24	46.63
-76.18	37.84
-80.54	28.50
-84.29	18.67
-87.39	8.44
-89.83	-2.13
-91.59	-12.95
-92.65	-23.93
-93.00	-35.00
-91.59	-35.87
-87.39	-36.71
-80.54	-37.50
-71.24	-38.21
-59.78	-38.83
-46.50	-39.33
-31.81	-39.70
-16.15	-39.92
-0.00	-40.00
17.00	-40.00
17.00	0.00
-17.00	0.00
-16.94	8.02
-16.74	15.98
-16.42	23.81
-15.97	31.47

Клин (2) – координати X та Y вершин другої деталі

-15.41	38.88
-14.72	46.00
-13.93	52.77
-13.02	59.14
-12.02	65.05
-10.93	70.48
-9.75	75.36
-8.50	79.67
-7.18	83.38
-5.81	86.45
-4.40	88.87
-2.95	90.60
-1.48	91.65
-0.00	92.00
8.11	91.52
16.15	90.07
24.07	87.67
31.81	84.34
39.30	80.10
46.50	74.99
53.34	69.03
59.78	62.29
65.76	54.80
71.24	46.63
76.18	37.84
80.54	28.50
84.29	18.67
87.39	8.44
89.83	-2.13
91.59	-12.95
92.65	-23.93
93.00	-35.00
91.59	-35.87
87.39	-36.71
80.54	-37.50
71.24	-38.21
59.78	-38.83
46.50	-39.33
31.81	-39.70
16.15	-39.92
0.00	-40.00
-17.00	-40.00
-17.00	0.00

-175.00	38.00	<i>Основа панамми</i> – координати X та Y вершин третьої деталі
-172.34	36.78	
-164.45	35.61	
-151.55	34.50	
-134.06	33.50	
-112.49	32.64	
-87.50	31.94	
-59.85	31.42	
-30.39	31.11	
-0.00	31.00	

.....	
175.00	-38.00
172.34	-36.78
164.45	-35.61
151.55	-34.50
134.06	-33.50
112.49	-32.64
87.50	-31.94
59.85	-31.42
30.39	-31.11
0.00	-31.00
-30.39	-31.11
-59.85	-31.42
-87.50	-31.94
-112.49	-32.64
-134.06	-33.50
-151.55	-34.50
-164.45	-35.61
-172.34	-36.78
-175.00	-38.00
-175.00	38.00

157.74	-78.17	<i>Поле панамми</i> – координати X та Y вершин четвертої деталі
146.92	-96.99	
133.86	-114.34	
118.77	-129.95	
101.87	-143.58	
83.42	-155.03	
63.70	-164.12	
43.01	-170.71	
21.67	-174.71	
0.00	-176.04	
-21.67	-174.71	
-43.01	-170.71	

.....	
-59.73	-111.00

-45.61	-117.50
-30.80	-122.22
-15.52	-125.09
0.00	-126.04
15.52	-125.09
30.80	-122.22
45.61	-117.50
59.73	-111.00
72.93	-102.80
85.03	-93.04
95.84	-81.86
105.19	-69.44
112.94	-55.97
157.74	-78.17

Висновки до третього розділу

1. Запропоновані алгоритми реалізовані в програмному забезпеченні для автоматизованого проєктування головних уборів. Це програмне забезпечення має дружній інтерфейс та не потребує допоміжних знань з комп'ютерних наук при роботі з ним.
2. Розроблене програмне забезпечення для автоматизованого проєктування головних уборів впроваджено у навчальному процесі на кафедрі інформаційних технологій проєктування при викладанні дисципліни «САПР при конструюванні виробів в галузях легкої промисловості».
3. Це програмне забезпечення для автоматизованого проєктування головних уборів може бути впроваджене на малих швейних підприємствах по індивідуальному пошивці головних уборів.

Висновки

1. Запропоновано алгоритми автоматизованого проєктування головних уборів, а саме: беретів, панам (звичайна панама, панама із широкими полями та молодіжна панама), конфедератки шляпи-циліндр та ковбойської шляпи.
2. Розроблено алгоритми допоміжних задач, які дозволили успішно реалізувати алгоритми автоматизованого проєктування головних уборів.
3. Запропоновані алгоритми реалізовані в програмному забезпеченні для автоматизованого проєктування головних уборів. Це програмне забезпечення має дружній інтерфейс та не потребує допоміжних знань з комп'ютерних наук при роботі з ним.
4. Розроблене програмне забезпечення для автоматизованого проєктування головних уборів впроваджено у навчальному процесі на кафедрі інформаційних технологій проєктування при викладанні дисципліни «САПР при конструюванні виробів в галузях легкої промисловості».
5. Це програмне забезпечення для автоматизованого проєктування головних уборів може бути впроваджене на малих швейних підприємствах по індивідуальному пошивці головних уборів.

Список використаних джерел

1. Гліненко Л.К., Сухонос О.Г. Основи моделювання технічних систем. Львів: «Бескид Біт», 2003. 176 с.
2. Томашевский В.М. Моделирование систем. К.: Видавнича група BVH, 2005. 352с.
3. Тимченко А.А. Основи системного проектування та системного аналізу складних об'єктів. Основи САПР та системного проектування складних об'єктів: Підручник для студ. вищ. навч. закл. /За ред.. В.І. Бикова. 2-е вид. – Київ: Либідь, 2003, 270 с.
4. Колосніченко М.В., Щербань В.Ю., Процик К.Л. Комп'ютерне проектування одягу: Навчальний посібник. – К.: “Освіта України”, 2010. – 236 с.
5. Чупринка В.І., Гаркавенко С.С., Чупринка Н.В. Інтерактивний метод проектування галантерейних виробів Київ, КНУТД- 2016. – 120 с.
6. Чупринка В.І., Гаркавенко С.С. Автоматизоване проектування раціональних схем розкрою рулонних матеріалів на деталі взуття та галантерейних виробів Київ, КНУТД- 2017. – 196 с.
7. Залкінд В. В. Визначення напрямків розвитку САПР одягу / В. В. Залкінд, М. Л. Рябчиков // Вісник Східноукраїнського національного університету імені Володимира Даля. – 2010. – №1, Ч.1 – С. 135–139.
8. Славінська А.Л. Методи типового проектування одягу: Навчальний посібник - Хмельницький: ХНУ, 2007. -159 с.
9. Процик, К.Л. Розробка технології проектування базових конструкцій одягу похідних покроїв на основі принципів трансформації: дис. ...канд.техн.наук: 05.19.04. – К., 2004. – 158 с.
10. Косенко О. І. Аналіз графічних матеріалів для проведення структурного аналізу костюму і художній системі формоутворення “колекція” / О. І. Косенко // Вісник Харківської державної академії дизайну і мистецтв. – 2011. – № 1. – С. 17–21

11. Пястук О.В., Бондар К.І. Застосування комп'ютерних технологій для вдосконалення технологічної підготовки швейного виробництва / О. В. Пястук, К. І. Бондар. – Хмельницький : ХНУ, 2009. – С. 159–161.
12. Кущевський М. О., Швець Г. С. Матеріалознавство швейного виробництва [Текст] : навч. посіб. / М. О. Кущевський, Г. С. Швець ; видав-ц. дім «Кондор». — Київ : Кондор, 2021. — 412 с.
13. Цимбал Т. В., Фатихова Ю. А. Удосконалення процесу проектування жіночого одягу з врахуванням форм тіла та його асиметричності [Текст] / Т. В. Цимбал, Ю. А. Фатихова // Вісник Київського національного університету технологій та дизайну. — 2014. — № 4 (78). — С. 97–103.
14. Славінська А. Л. Методи типового проектування одягу: Навчальний посібник / А. Л.Славінська. – Хмельницький: ХНУ, 2012. – 179 с.
15. Славінська А. Л. Основи модульного проектування одягу: Монографія / А. Л.Славінська. – Хмельницький: ХНУ, 2007. – 167 с.
16. Пястук О.В., Бондар К.І. Застосування комп'ютерних технологій для вдосконалення технологічної підготовки швейного виробництва / О. В. Пястук, К. І. Бондар. – Хмельницький : ХНУ, 2009. – С. 159–161.
17. Залкінд В. В. Особливості розробки конструкцій одягу з урахуванням можливостей сучасних САПР / В. В. Залкінд // Вісник Київського національного університету технологій та дизайну. – 2010. – №1. – С. 99–103.
18. Клопотівська Я. М. Особливості дизайн-проектування аксесуарів з метою коригування форми обличчя / Я. М. Клопотівська, М. А. Говда, М. С. Винничук // Тези доповідей XVII Всеукраїнської наукової конференції молодих вчених та студентів [«Наукові розробки молоді на сучасному етапі»] (27-28 квітня 2018 р.) М-во освіти і науки України, КНУТД. – К. : КНУТД, 2018. – С. 131-132.
19. Буханцова Л. В. Морфологічні типи голів як вихідні дані для проектування жіночих головних уборів / Л. В. Буханцова, О. П. Стрижова, М. О. Кущевський // Вісник Хмельницького національного університету. – 2005. – №1 (65). – С. 123-126.

20. Ергономіка і дизайн. Проектування сучасних видів одягу: Навчальний посібник / [Колосніченко М. В., Зубкова Л. І., Пашкевич К. Л. та ін.]. – К.: НВЦ "Профі", 2014. – 386 с.
21. Розробка колекцій одягу: Навчальний посібник / А. М. Малинська, К. Л. Пашкевич, М. Р. Смирнова, О. В. Колосніченко – К.: ПП НВЦ Профі, 2014. – 140 с.
22. Васильєва О.С. Система антропометричних ознак голови та її вплив на дизайн головних уборів. Прикладна геометрія та інженерна графіка. Праці / Таврійський державний агротехнічний університет –Вип. 4, т.40. - Мелітополь:2008,164 с.
23. Васильєва О.С. Вікові особливості та зміни морфології голови людини. Вісник Хмельницького національного університету 6,2008.
24. Чупринка.В.І. Підготовка інформації про зовнішні контури плоских геометричних об'єктів для для ефективного автоматизованого проектування раціональних схем розкрою / В.І. Чупринка, Н.В. Чупринка, Б.В Науменко, О.Л. Василенко //Технології та інжиніринг, № 4(9), 2022. С.48-59
25. В.П. Яковець, В.Н. Боровик, А.В. Ваврикович Аналітична геометрія/ КНУ.2020. 298 с.
26. Осадча Л. К. Лінійна алгебра та аналітична геометрія : навч. посіб. / Л. К. Осадча ; Нац. ун-т вод. госп-ва та природокористування. – Рівне : НУВГП, 2020. – 148 с.
27. Аналітична геометрія Підручник для вищих навчальних закладів Б. В. Гринев, І. К . Кириченко Гімназія 344 с.
28. Є. П. Зайцев: Вища математика. Лінійна та векторна алгебра, аналітична геометрія, вступ до математичного аналізу Алерта 574 с.
29. Ковалюк Т.В. Основи програмування: Підручник. / Т.В. Ковалюк, М.З. Згуровський – К: ВНУ, 2005. – 375 с.
30. Шаховська, Н. Б. Алгоритми і структури даних: посібник / Н.Б. Шаховська, Р.О. Голощук; - Львів: Магнолія, 2010 – 215с.

31. Караванова Т. П. Основи алгоритмізації та програмування. Форум. – К.: 2002. – 286 с.
32. Ковалюк Т. В. Алгоритмізація та програмування: підручник з грифом МОН України / Т.В. Ковалюк. – Львів : Магнолія-2006, 2013. – 400 с.
33. Пустюльга С. І., Самчук В. П., Воробчук М. С. Інженерна та комп'ютерна графіка : навч. посіб. / С. І. Пустюльга, В. П. Самчук, М. С. Воробчук ; Луцький нац. техн. ун-т. — Луцьк : Просто Друк, 2024. — Частина 1. — 324 с

Публікація за темою кваліфікаційної роботи

SCI-CONF.COM.UA

CURRENT TRENDS IN SCIENTIFIC RESEARCH DEVELOPMENT



**PROCEEDINGS OF III INTERNATIONAL
SCIENTIFIC AND PRACTICAL CONFERENCE
OCTOBER 17-19, 2024**

**BOSTON
2024**

CURRENT TRENDS IN SCIENTIFIC RESEARCH DEVELOPMENT

Proceedings of III International Scientific and Practical Conference

Boston, USA

17-19 October 2024

Boston, USA

2024

2

24.	<i>Чупринка В. І., Чупринка Н. В., Шегеда Д. С.</i> ПРОГРАМНИЙ ПРОДУКТ ДЛЯ АВТОМАТИЗОВАНОГО ПРОЄКТУВАННЯ ГОЛОВНИХ УБОРІВ	152
25.	<i>Ясечко М. М., Чеканов А. В., Рязанцев С. С., Ворошилов К. А.</i> ПРОГРАМНИЙ КОМПЛЕКС-ТРЕНАЖЕР ОБСЛУГИ САМОХІДНОЇ УСТАНОВКИ РОЗВІДКИ ТА НАВЕДЕННЯ ІС91 ЗЕНІТНОГО РАКЕТНОГО КОМПЛЕКСУ "КУБ"	161
PHYSICAL AND MATHEMATICAL SCIENCES		
26.	<i>Pysarenko A. M.</i> PROPAGATION OF ULTRASONIC SIGNALS IN A POROUS COMPOSITE	165
27.	<i>Yaltychenko O. V., Kanarovskii E. Yu.</i> ELECTRON TRANSFER IN TRIMER NANOCUSTER INTERACTED WITH ORGANIC MATRIX IN ELECTRIC FIELD	169
28.	<i>Алісова В. Г.</i> ВИВЧЕННЯ ГЕОМЕТРИЧНОЇ ЙМОВІРНОСТІ ЗДОБУВАЧАМИ ОСВІТИ	180
GEOGRAPHICAL SCIENCES		
29.	<i>Ознамець О. Й., Зелінська С. В.</i> ВИКОРИСТАННЯ МОДЕЛЕЙ І СХЕМ В ЕКОЛОГІЧНОМУ ВИХОВАННІ ДОШКІЛЬНИКІВ	186
GEOLOGICAL AND MINERALOGICAL SCIENCES		
30.	<i>Вуцневич І. В., Тузук Ya. М.</i> VERTEBRATE FOOTPRINTS FROM THE LVTV COLLECTION: COLOR- INTENSITY ANALYSIS OF PHOTOGRAPHIC IMAGES	195
ARCHITECTURE		
31.	<i>Войко Н. Ю., Пешкова О. О.</i> ЗАСОБИ ВИРАЖЕННЯ ХАОСУ В ЛАНДШАФТНО-ПЛАНУВАЛЬНІЙ ОРГАНІЗАЦІЇ КУЛЬТУРНО-ПІЗНАВАЛЬНОГО ПАРКУ ЧАСУ	201
32.	<i>Тігарєва Т. Г.</i> ПРОБЛЕМИ ВІДНОВЛЕННЯ БУДІВЕЛЬ І СПОРУД, ЩО ПОШКОДЖЕНІ В РЕЗУЛЬТАТІ ВОЄННИХ ДІЙ	208
PEDAGOGICAL SCIENCES		
33.	<i>Белый С. П.</i> АСПЕКТИ ПРОФЕСІЙНОЇ ДІЯЛЬНОСТІ ТРЕНЕРА З ПЛАВАННЯ	211
34.	<i>Гірняк С. П., Сарабаха Г. Б.</i> ТРАДИЦІЙНІ ТА НЕТРАДИЦІЙНІ ФОРМИ ОРГАНІЗАЦІЇ ОСВІТНЬОГО ПРОЦЕСУ В ПОЧАТКОВИХ КЛАСАХ: ПЕРЕВАГИ І НЕДОЛІКИ	217

УДК 687.04.02

ПРОГРАМНИЙ ПРОДУКТ ДЛЯ АВТОМАТИЗОВАНОГО ПРОЄКТУВАННЯ ГОЛОВНИХ УБОРІВ

Чупринка Віктор Іванович

д.т.н, проф.

Чупринка Наталія Вікторівна

к.т.н, доц.

Шегеда Дмитро Сергійович

Студент

Київський національний університет технологій та дизайну

м. Київ, Україна

Анотація: В роботі запропонований програмний продукт для автоматизованого проєктування головних уборів, а саме беретів, панам (звичайна панама, панама із широкими полями та молодіжна панама), конфедератки, шляпи-циліндр та ковбойської шляпи. Програмний продукт має дружній інтерфейс і не потребує додаткових знань з області комп'ютерних наук.

Ключові слова: програмний продукт, автоматизоване проєктування, головні убори.

В реалізації поставлених перед легкою промисловістю задач по збільшенню виробництва, розширенню асортименту, та покращення якості товарів народного вжитку значне місце займає розробка та залучення сучасних методів та засобів дослідів, а також контролю якості виробів.

При проєктуванні нового виробу, вирішальне значення мають процеси технічного моделювання і конструювання.

Під технічним (конструктивним) моделюванням мають на увазі інженерний процес розробки рисунку або лекал деталей виробу за первинним зразком нової моделі або за її графічним зображенням з використанням рисунків чи лекал деталей відповідної базової основи конструкції

Якість конструювання й моделювання виробів - один з найбільш важливих факторів швидкої реалізації й високого попиту на продукцію швейного підприємства. Автоматизація конструювання головних уборів дозволяє прискорити розробку нових моделей, скоротити число рутинних операцій, підвищити якість посадки, відповідність розмірним ознакам. Вона є необхідним елементом САПР сучасного швейного підприємства.

Створення високоефективного методу отримання інформації та проєктування головних уборів з використанням сучасних технічних засобів дозволить суттєво зменшити термін виконання робіт, підвищити продуктивність праці, поліпшити якість проектних рішень та ефективність виробництва.

Для автоматизованого проєктування головних уборів були вибрані наступні моделі головних уборів: берет, панамі (звичайна панاما, панاما із широкими полями та молодіжна панاما), конфедератка, шляпа-циліндр та ковбойська шляпа. При створенні програмного продукту проєктування головних уборів були використана практика ручного проєктування цих виробів [1-2], на основі якої були розроблені алгоритми для автоматизованого проєктування деталей для кожної із запропонованої моделі головного убору за індивідуальними параметрами голови людини, для якої проєктується головний убір та технологічними параметрами головного убору.

При розробці алгоритмів для автоматизованого проєктування головних уборів виникли деякі задачі, які необхідно було вирішити. Розглянемо одну із них. Це побудова відрізка заданої величини на прямій, перпендикулярній заданій прямій, що проходить через задану точку

Дано: Пряма AB , що задана двома точками: $A(X_A; Y_A)$, $B(X_B; Y_B)$. На прямій AB задана точка $C(X_C; Y_C)$.

З точки C провести відрізок, перпендикулярний прямій AB довжиною Δ (рис. 1).

Знайти: координати точки $D(X_D; Y_D)$.

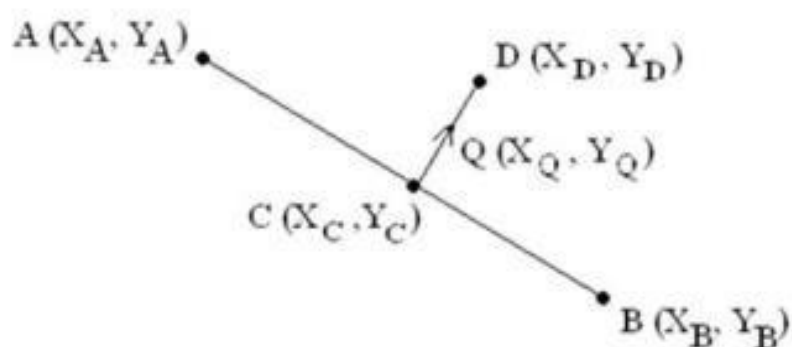


Рис. 1. Побудова відрізка заданої величини на прямій

Рівняння прямої AB , що проходить через дві точки можна представити наступним чином[3]:

$$\frac{X - X_A}{X_B - X_A} = \frac{Y - Y_A}{Y_B - Y_A} \Rightarrow Ax + By + C = 0;$$

$$A = Y_B - Y_A;$$

$$B = X_A - X_B;$$

$$C = X_B \cdot Y_A - X_A \cdot Y_B.$$

Можливі три випадки:

а) Пряма AB паралельна вісі OX (рис. 2).

$$A = 0; By + C = 0 \Rightarrow Y = -\frac{C}{B}; Y_A = Y_B = Y_C = -\frac{C}{B}.$$

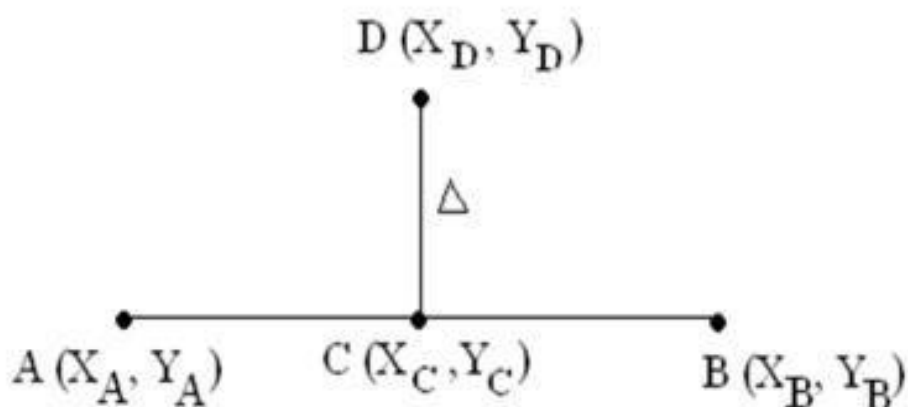


Рис. 2. Пряма AB паралельна вісі OX

Тоді $X_D = X_C$.

$$Y_D = Y_A \pm \Delta = -\frac{C}{B} \pm \Delta.$$

б) Пряма AB паралельна вісі OY (рис. 3).

$$B = 0; Ax + C = 0 \Rightarrow X = -\frac{C}{A}; X_A = X_B = X_C = -\frac{C}{A};$$

$$X_D = X_C \pm \Delta = -\frac{C}{A} \pm \Delta; Y_D = Y_C.$$

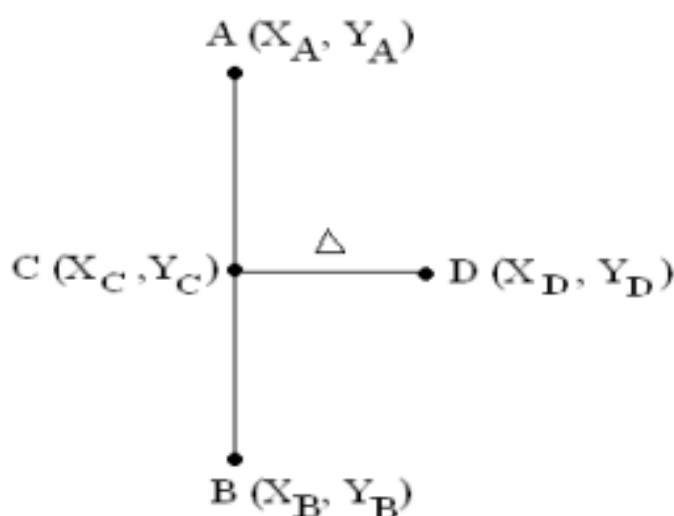


Рис. 3. Пряма AB паралельна вісі OY

в) Загальний випадок (пряма AB не паралельна осям) (рис.4).

Тоді рівняння прямої $CD \perp AB$ можна представити наступним чином [1]:

$$CD: Y - Y_C = K_1 \cdot (X - X_C); \text{ де } K_1 \cdot K = -1 \Rightarrow K_1 = -\frac{1}{K}. \text{ Визначимо } K:$$

$$Ax + By + C = 0 \Rightarrow By = -Ax - C \Rightarrow Y = \frac{-Ax - C}{B} = -\underbrace{\frac{A}{B}}_K X - \frac{C}{B} = KX - \frac{C}{B}$$

$$K = -\frac{A}{B}$$

Рівняння прямої CD має наступний вигляд: $Y - Y_C = \frac{X - X_C}{K}$ або

$$Y = Y_c + \frac{X - X_c}{K};$$

Визначимо довільну точку Q на прямій CD . Нехай: $X_Q = X_c \pm \Delta q$;

$$\text{Тоді: } Y_q = Y_c + \frac{X_q - X_c}{K} = Y_c \pm \frac{\Delta q}{K};$$

Тоді координати вектору $\overrightarrow{CQ}$ визватимуться наступним чином:

$$\overrightarrow{CQ} = \{X_Q - X_c; Y_Q - Y_c\};$$

Визначимо одиничний вектор на прямій CD : $\vec{e}_{CQ} = \left\{ \frac{X_Q - X_c}{D}; \frac{Y_Q - Y_c}{D} \right\}$,

$$\text{де } D = |\overrightarrow{CQ}| = \sqrt{(X_Q - X_c)^2 + (Y_Q - Y_c)^2}.$$

$$\overrightarrow{CD} = \{X_D - X_c; Y_D - Y_c\} = \Delta \cdot \vec{e}_{CQ} = \left\{ \frac{\Delta}{D} (X_Q - X_c); \frac{\Delta}{D} (Y_Q - Y_c) \right\};$$

$$\text{Звідси: } \begin{cases} X_D - X_c = \frac{\Delta}{D} (X_Q - X_c) \\ Y_D - Y_c = \frac{\Delta}{D} (Y_Q - Y_c) \end{cases} \Rightarrow \begin{cases} X_D = X_c + \frac{\Delta}{D} (X_Q - X_c) \\ Y_D = Y_c + \frac{\Delta}{D} (Y_Q - Y_c) \end{cases}$$

Використавши запропоновані алгоритми автоматизованого проєктування головних уборів за індивідуальними розмірними ознаками голови людини, для якої проєктується головний із асортименту запропонованих головних уборів, був розроблений програмний продукт для автоматизованого проєктування головних уборів.

Даний програмний продукт було розроблено у середовищі програмування Delphi 7.0. Середовищі програмування Delphi 7.0 було вибране тому, що воно включає не тільки мову програмування, а й ефективне діалогове середовище для розробки форм і вікон, що дозволяє програмісту створити програмний продукт достатньо високої якості за порівняно невеликий проміжок часу. Delphi 7.0 дає можливість використовувати ресурси інших прикладних програм Windows.

Структурна схема програмного продукту представлена на рис. 1.

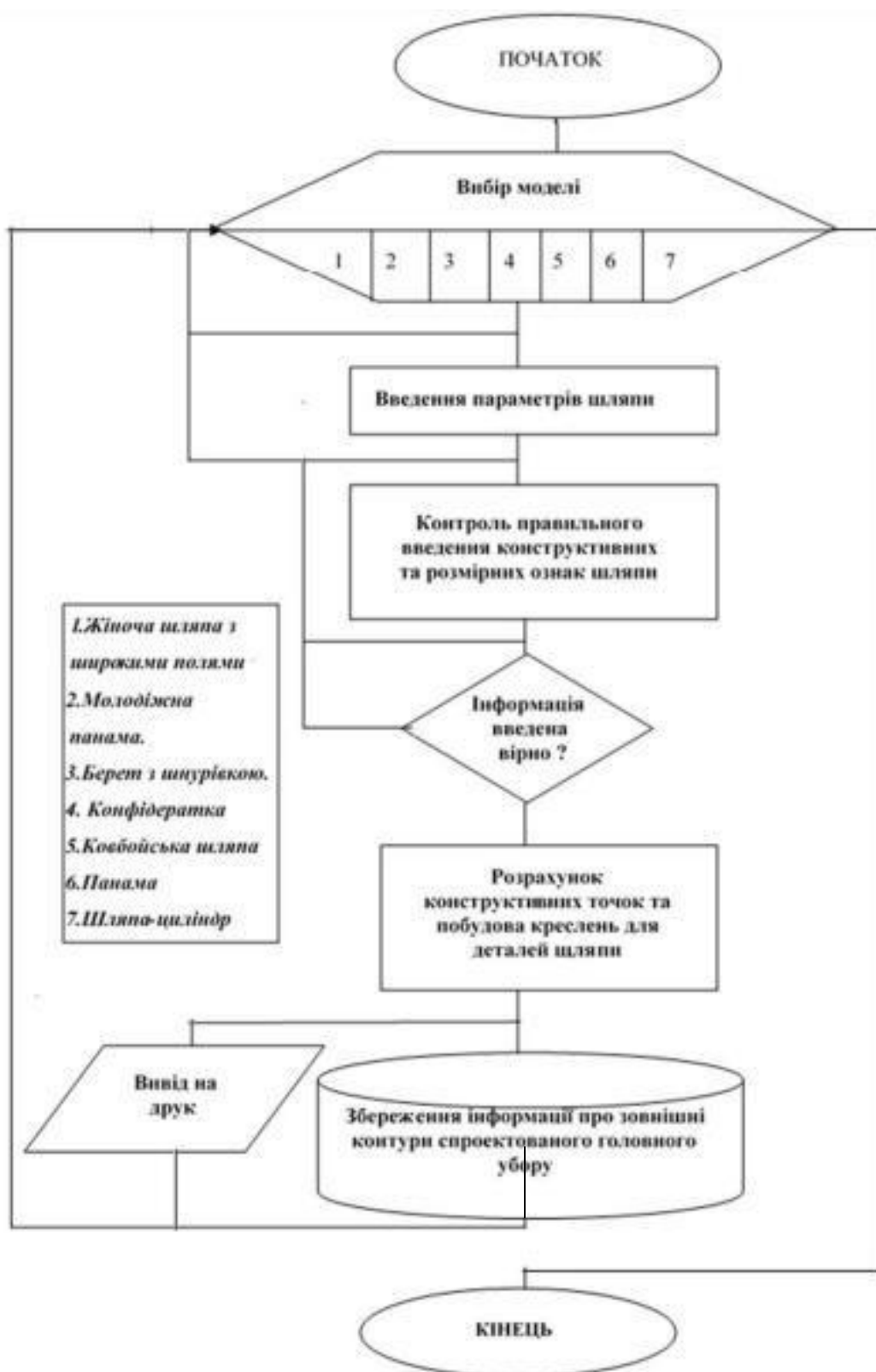


Рис. 1. Структурна схема програмного продукту

На рис. 2 представлена основна форма програмного продукту для автоматизованого проектування головних уборів.



Рис. 2. Основна форма програмного продукту

Після вибору однієї із моделей головного убрання, ми переходимо до закладки для введення основних розмірів головного убрання (рис. 3).

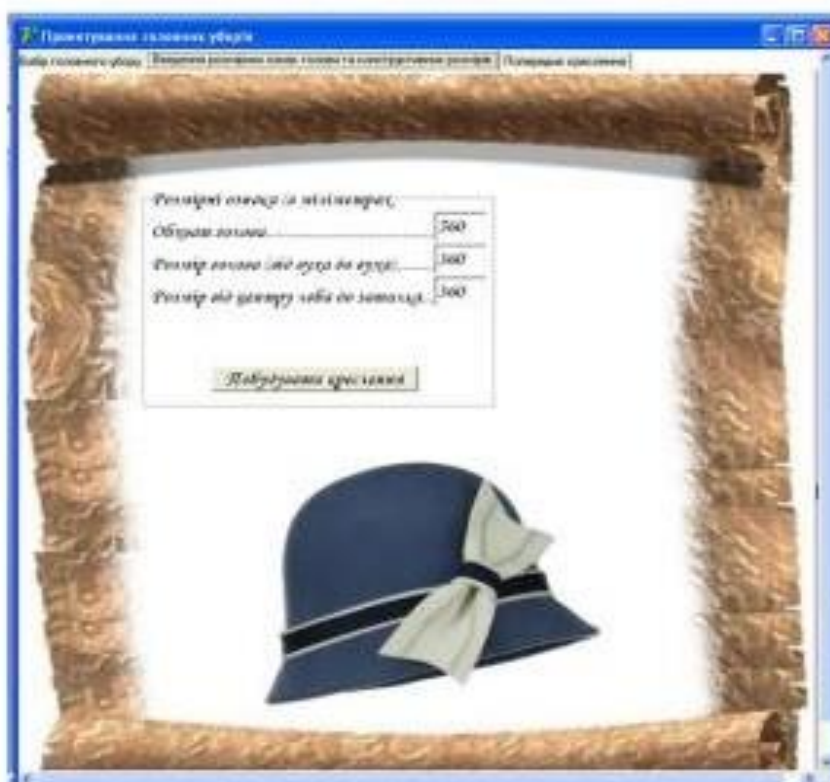


Рис. 3. Введення розмірних ознак голови

Висновки

1. Запропоновано алгоритми автоматизованого проєктування беретів, панам (звичайна панама, панама із широкими полями та молодіжна панама), конфедератки шляпи-циліндр, та ковбойської шляпи.
2. Розроблено алгоритми допоміжних задач, які дозволили успішно реалізувати алгоритми автоматизованого проєктування головних уборів.
3. Запропоновані алгоритми реалізовані в програмному забезпеченні для автоматизованого проєктування головних уборів, що має дружній інтерфейс та не потребує допоміжних знань з комп'ютерних наук при роботі з ним.
4. Розроблене програмне забезпечення для автоматизованого проєктування головних уборів впроваджено у навчальному процесі на кафедрі інформаційних технологій проєктування при викладанні дисципліни «САПР при конструюванні виробів в галузях легкої промисловості».
5. Розроблене програмне забезпечення для автоматизованого проєктування головних уборів може бути впроваджене на малих швейних підприємствах по індивідуальному пошивці головних уборів.

СПИСОК ЛІТЕРАТУРИ

1. Васильєва О.С. Система антропометричних ознак голови та її вплив на дизайн головних уборів. Прикладна геометрія та інженерна графіка. Праці / Таврійський державний агротехнічний університет – Вип. 4, т. 40. - Мелітополь:2008,164 с.
2. Васильєва О.С. Вікові особливості та зміни морфології голови людини. Вісник Хмельницького національного університету 6,2008
3. В.П. Яковець, В.Н. Боровик, А.В. Ваврикович Аналітична геометрія/ КНУ shop.2020. 298 с.
4. Михайленко В. Є. Інженерна та комп'ютерна графіка. К. : Вища школа, 2002. 332 с.



МІНІСТЕРСТВО
ОСВІТИ І НАУКИ
УКРАЇНИ

ХНТУ
КИЇВСЬКИЙ НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ
ТЕХНОЛОГІЙ ТА ДИЗАЙНУ

СЕРТИФІКАТ УЧАСНИКА

ВСЕУКРАЇНСЬКОГО КОНКУРСУ НАУКОВИХ ДОСЛІДЖЕНЬ СТУДЕНТІВ
"ІНФОРМАЦІЙНІ ТЕХНОЛОГІЇ В НАУЦІ ТА ВИРОБНИЦТВІ"

Шегеда Дмитро

СТУДЕНТ КИЇВСЬКОГО НАЦІОНАЛЬНОГО УНІВЕРСИТЕТУ
ТЕХНОЛОГІЙ ТА ДИЗАЙНУ

НАЗВА РОБОТИ " «АЛГОРИТМІЧНЕ ТА ПРОГРАМНЕ ЗАБЕЗПЕЧЕННЯ
ДЛЯ АВТОМАТИЗОВАНОГО ПРОЄКТУВАННЯ ДЕТАЛЕЙ ГОЛОВНИХ
УБОРІВ » ПІД ШИФРОМ «ГОЛОВНІ УБОРИ» "

19-20 березня 2024



ректор ХНТУ, д.т.н., професор
Олена ЧЕПЕЛЮК

ВСЕУКРАЇНСЬКИЙ КОНКУРС
НАУКОВИХ ДОСЛІДЖЕНЬ СТУДЕНТІВ

Листинг программного продукта

nit UnDgGolUb;

interface

uses

*Windows, Messages, SysUtils, Variants, Classes, Graphics, Controls, Forms,
Dialogs, ComCtrls, StdCtrls, jpeg, ExtCtrls, Printers;*

type

TForm1 = class(TForm)

Image14: TImage;

Button1: TButton;

PageControl1: TPageControl;

TabSheet1: TTabSheet;

TabSheet2: TTabSheet;

Image2: TImage;

Label3: TLabel;

Label4: TLabel;

Label5: TLabel;

GroupBox1: TGroupBox;

Label1: TLabel;

Label2: TLabel;

Button2: TButton;

Edit1: TEdit;

Edit2: TEdit;

Label6: TLabel;

Edit3: TEdit;

Edit4: TEdit;

Label7: TLabel;

TabSheet3: TTabSheet;

Image5: TImage;

Image6: TImage;

Image8: TImage;

Image10: TImage;

Image3: TImage;

Image7: TImage;

Image9: TImage;

Image11: TImage;

GroupBox2: TGroupBox;

Edit5: TEdit;

Label8: TLabel;


```

Edit6: TEdit;
Label9: TLabel;
Edit7: TEdit;
Edit8: TEdit;
Label10: TLabel;
Label11: TLabel;
Button3: TButton;
SaveDialog1: TSaveDialog;
Button4: TButton;
PrintDialog1: TPrintDialog;
Image1: TImage;
Image4: TImage;
Image12: TImage;
procedure FormCreate(Sender: TObject);
procedure Button1MouseDown(Sender: TObject; Button: TMouseButton;
  Shift: TShiftState; X, Y: Integer);
// procedure Image2MouseDown(Sender: TObject; Button: TMouseButton;
//   Shift: TShiftState; X, Y: Integer);
// procedure Image3MouseDown(Sender: TObject; Button: TMouseButton;
//   Shift: TShiftState; X, Y: Integer);
// procedure Image7MouseDown(Sender: TObject; Button: TMouseButton;
//   Shift: TShiftState; X, Y: Integer);
// procedure Image8MouseDown(Sender: TObject; Button: TMouseButton;
//   Shift: TShiftState; X, Y: Integer);
// procedure Image9MouseDown(Sender: TObject; Button: TMouseButton;
//   Shift: TShiftState; X, Y: Integer);
procedure Edit1Click(Sender: TObject);
procedure Edit2Click(Sender: TObject);
procedure Edit3Click(Sender: TObject);
procedure Button2Click(Sender: TObject);
// procedure Image6MouseDown(Sender: TObject; Button: TMouseButton;
//   Shift: TShiftState; X, Y: Integer);
procedure Image10MouseDown(Sender: TObject; Button: TMouseButton;
  Shift: TShiftState; X, Y: Integer);
procedure Button1Click(Sender: TObject);
procedure Button3Click(Sender: TObject);
procedure Button4Click(Sender: TObject);
procedure Button5Click(Sender: TObject);
procedure GroupBox2Click(Sender: TObject);

// procedure Image12MouseDown(Sender: TObject; Button: TMouseButton;
//   Shift: TShiftState; X, Y: Integer);
// procedure TabSheet3ContextPopup(Sender: TObject; MousePos: TPoint;
//   var Handled: Boolean);

```

```

private
  { Private declarations }
public
  { Public declarations }
end;

var
  Form1: TForm1;

ObhGol,GlGolV,GlGolL,GlGolR,n,n_kl,n_kl2,n_kl0,n_osn,Npol,NklNew,col_det:integer;

PrEdit1,PrEdit2,PrEdit3,PrEdit4,Pr_Im2,Pr_Im3,Pr_Im6,Pr_Im7,Pr_Im8,Pr_Im9,Pr_Im11:Boolean;
alfa,alfa1,DovR,DovKlN,Teta,Xo,Yo,Xc1,Yc1,Xc2,Yc2,Xd1,Yd1,Xd2,Yd2,H,deltaR,delta,delta1,MXY_im5:real;
  X_kl,Y_kl,x_kl2,y_kl2,x_kl2z,y_kl2z,x_osn,y_osn,
Xpol,Ypol,X_KL0,Y_KL0,X_pol,Y_pol,
XklNew,YKlNew:array[0..100] of real;
x_det,y_det:array[1..10,0..100]of real;
name_det:array[1..20] of string;
col_point_det:array[1..10]of integer;
model:string[20];

implementation

uses DateUtils;

{$R *.dfm}

procedure TForm1.FormCreate(Sender: TObject);
begin
  {
  GroupBox2.Brush.Style:=bsClear;
  SetWindowLong(GroupBox2.Handle,
  GWL_EXSTYLE,GetWindowLong(GroupBox2.Handle, GWL_EXSTYLE) or
  WS_EX_TRANSPARENT);
  GroupBox1.Brush.Style:=bsClear;
  SetWindowLong(GroupBox1.Handle,
  GWL_EXSTYLE,GetWindowLong(GroupBox1.Handle, GWL_EXSTYLE) or
  WS_EX_TRANSPARENT);    }

```

```
//InvalidateRect(GroupBox2.Handle, NiLL, true);
```

```
Width:=1000;//800;
Height:=800;//600;
Left:=20;
Top:=0;
PrEdit1:=True;
PrEdit2:=True;
PrEdit3:=True;
PrEdit4:=True;
deltaR:=23;
delta:=16;
delta1:=10;
end;
```

```
Procedure SmeshchenieVector(Xa,Ya,Xb,Yb,delta:real;var Xc,Yc:real);
var Dx,Dy,D,Edx,Edy:real;
begin
  Dx:=Xb-Xa; Dy:=Yb-Ya;
  D:=sqrt(sqr(Dx)+sqr(Dy));
  Edx:=Dx/D;
  Edy:=Dy/D;
  Xc:=Xb+delta*Edx;
  Yc:=Yb+delta*Edy;
end;
```

```
Procedure CrossTwoLine(X1,Y1,X2,Y2,X3,Y3,X4,Y4:real;var X,Y:real);
var delta1,delta2,delta,A1,A2,B2,B1,C1,C2:real;
begin
  A1:=y1-y2; A2:=y3-y4;
  B1:=x2-x1; B2:=x4-x3;
  C1:=x1*y2-x2*y1; C2:=x3*y4-x4*y3;
  delta:=A1*B2-A2*B1;
  delta1:=B1*C2-B2*C1;
  delta2:=A1*C2-A2*C1;
  X:=delta1/delta;
  Y:=delta2/delta;
end;
```

```

Procedure Perpend(Xa,Ya,Xb,Yb,delta:real;var Xc1,Yc1,Xc2,Yc2:real);
var
k,Xd1,Yd1,Xd2,Yd2,Dbd:real;
begin
  if Xa=Xb then
    begin
      Xc1:=Xb+delta;
      Xc2:=Xb-delta;
      Yc1:=Yb;
      Yc2:=Yb;
    end
  else
    if Ya=Yb then
      begin
        Yc1:=Yb+delta;
        Yc2:=Yb-delta;
        Xc1:=Xb;
        Xc2:=Xb;
      end
    else
      begin
        k:=(Yb-Ya)/(Xb-Xa);
        Xd1:=Xb+5;
        Xd2:=Xb-5;
        Yd1:=Yb-(Xd1-Xb)/k;
        Yd2:=Yb-(Xd2-Xb)/k;
        SmeshchenieVector(Xd1,Yd1,Xb,Yb,delta,Xc1,Yc1);
        SmeshchenieVector(Xd2,Yd2,Xb,Yb,delta,Xc2,Yc2);
      end;
    end;
end;

```

```

procedure TForm1.Button1MouseDown(Sender: TObject; Button: TMouseButton;
  Shift: TShiftState; X, Y: Integer);
begin
  PageControl1.Visible:=true;
  PageControl1.ActivePage:=TabSheet1;
  TabSheet2.Visible:=False;
  TabSheet2.TabVisible:=False;

```


end;

```
procedure TForm1.Edit1Click(Sender: TObject);
begin
  Image4.Picture.LoadFromFile('Y1.bmp');
end;
```

```
procedure TForm1.Edit2Click(Sender: TObject);
begin
  if Pr_Im7 then Form1.Image4.Visible:=False else
  begin
    Form1.Image4.Visible:=True;
    Image4.Picture.LoadFromFile('Y3.bmp');
  end;
end;
```

```
procedure TForm1.Edit3Click(Sender: TObject);
begin
  if Pr_Im7 then Form1.Image4.Visible:=False else
  begin
    Form1.Image4.Visible:=True;
    Image4.Picture.LoadFromFile('Y4.bmp');
  end;
end;
```

```
Function Max_Min(n,p:integer; z:array of real ):real;
var
  i:integer;
  MM:real;
begin
  MM:=z[0];
  for i:=1 to n-1 do
    if MM*p<z[i]*p
    then
      MM:=z[i];
      Max_Min:=MM;
  end;
```

```
Procedure Graf_im5(n:integer;x,y: array of real;
MXY,Xcf,Ycf,XcE,YcE:real;P,Q,T:integer);
var
  i:integer;
```

X_r, Y_r: array[0..100] of integer;

```

begin
  for i:=0 to n-1 do
    begin
      X_r[i]:=Round((x[i]-Xcf)*MXY+XcE);
      Y_r[i]:=Round((-Y[i]+Ycf)*MXY+YcE);
    end;

    for i:=0 to n-1 do
      with Form1.Image5.Canvas do
        begin
          if T=1 Then
            Pen.Width:=1
          else Pen.Width:=3;

          if P=1 Then
            Pen.Color:=clBlack
          else
            Pen.Color:=clBlue;
          if i=0 then
            MoveTo(X_r[i],Y_r[i])
          else LineTo(X_r[i],Y_r[i])
          end;

          if Q=1 Then
            for i:=0 to n-1 do
              with Form1.Image5.Canvas do
                begin
                  Pen.Color:=ClRed;
                  Ellipse(X_r[i]-2,Y_r[i]-2,X_r[i]+2,Y_r[i]+2);
                end;
            end;
          end;
        end;
      end;
    end;
  end;

```

Procedure Elips(x,y, MXY,Xcf,Ycf,XcE,YcE:real);

```

var
  i,x_r,y_r:integer;
Begin

  X_r:=Round((x-Xcf)*MXY+XcE);
  Y_r:=Round((-Y+Ycf)*MXY+YcE);

  with Form1.Image5.Canvas do
    begin

```

```

Pen.Color:=ClRed;
  Ellipse(X_r-2,Y_r-2,X_r+2,Y_r+2);
end;
end;

```

```

  Procedure DugaElips(xc,yc,a,b,Fi1,Fi2:real; n:integer; var xr,yr:array of real);
  Var
  i:integer;
  Fi:real;
  x2,y2:array of real;
  Begin
  Fi:=(Fi2-Fi1)/n;
  for i:=0 to n do
  begin
  xr[i]:=xc+a*cos(Fi1+Fi*i);
  yr[i]:=yc+b*sin(Fi1+Fi*i);
  end;
  end;

```

```

  Procedure Duga(X0,Y0,R,AlfaB,AlfaE:real; var n:integer;var Xd,Yd:array of real);
  Var i:integer;
  D_Alfa:real;
  begin
  D_alfa:=AlfaE-AlfaB;
  n:=18;
  for i:=0 to n do
  begin
  Xd[i]:=R*cos(AlfaB+D_alfa*i/n)+x0;
  Yd[i]:=R*sin(AlfaB+D_alfa*i/n)+y0;
  end;
  n:=n+1;
  end;

```

```

  Procedure DetNewNumPointR(Var Nb,Ne:integer; Var kp:integer;
  Var xdet,ydet:array of real);
  Var D1,D2:real;
  i,Nr,kol:integer;
  Xr,Yr:array [0..100]of real;
  Xc,Yc:Int64;
  Begin
  If Nb>Ne then
  begin
  Nr:=Nb;

```

```

Nb:=Ne;
Ne:=Nr
end;
D1:=0;
For i:=Nb to Ne-1 do
  D1:=D1+Sqrt(sqr(Xdet[i+1]-Xdet[i])+sqr(Ydet[i+1]-Ydet[i]));
D2:=0;
For i:=0 to Nb-1 do
  D2:=D2+Sqrt(sqr(Xdet[i+1]-Xdet[i])+sqr(Ydet[i+1]-Ydet[i]));
For i:=Ne to KP-2 do
  D2:=D2+Sqrt(sqr(Xdet[i+1]-Xdet[i])+sqr(Ydet[i+1]-Ydet[i]));
If D1<D2 then
begin
  kol:=Ne-Nb+1;
  for i:=Ne to kp-1 do
  begin
    Xr[i-Ne]:=Xdet[i];
    Yr[i-Ne]:=Ydet[i];
  end;
  for i:=1 to Ne do
  begin
    Xr[kp-Ne-1+i]:=Xdet[i];
    Yr[kp-Ne-1+i]:=Ydet[i];
  end;
end
else
begin
  kol:=Kp-Ne+Nb;
  for i:=Nb to Kp-1 do
  begin
    Xr[i-Nb]:=Xdet[i];
    Yr[i-Nb]:=Ydet[i];
  end;
  for i:=1 to Nb do
  begin
    Xr[Kp-Nb-1+i]:=Xdet[i];
    Yr[Kp-Nb-1+i]:=Ydet[i];
  end;
end;
end;
// Button3.Caption:=IntToStr(kol);
Nb:=Kp-kol;
Ne:=Kp-1;
{  XcYc(kp,xr,yr,xc,yc);
  Graph_im2(1,mxy,kp,xr,yr,round(image2.Width/2),round(image2.Height/2),
xc,yc);}

```

```

for i:=0 to Kp-1 do
begin
  Xdet[i]:=Xr[i];
  Ydet[i]:=Yr[i];
end;
End;

```

```

Procedure Pole(Alf,ObhGol:real; var l,Fi:real);
var R:real;
begin
  R:=ObhGol/2/pi;
  l:=R/sin(Alf/2);
  Fi:=pi*R/l;
//end;
End;

```

```

Procedure GenPole(L,H,Fi:real;var Npol:integer; var Xpol,Ypol:array of real);
var i:integer;
  Fi0,DeltaFi:real;
begin
  Fi0:=-pi/2+fi/2;
  DeltaFi:=Fi/18;
  for i:=0 to 18 do
  begin
    Xpol[i]:=(L+H)*cos(Fi0-i*DeltaFi);
    Ypol[i]:=(L+H)*sin(Fi0-i*DeltaFi);
  end;
  Fi0:=-pi/2-Fi/2;
  for i:=0 to 18 do
  begin
    Xpol[i+19]:=L*cos(Fi0+i*DeltaFi);
    Ypol[i+19]:=L*sin(Fi0+i*DeltaFi);
  end;
  Xpol[38]:=Xpol[0];
  Ypol[38]:=Ypol[0];
  Npol:=39;
End;

```

```

Procedure PeretvorKluna(Nkl:integer; X_kl,Y_kl: array of real; h:real;var
NklNew:integer;var XklNew,YKlNew:array of real);
Var i:integer;
d:real;
begin
d:=Abs(X_kl[59]-X_kl[0]);

```

```

for i:=0 to 29 do
begin
XklNew[i]:=X_kl[i];
YklNew[i]:=Y_kl[i];
end;
for i:=29 to 59 do
begin
XklNew[i+4]:=X_kl[i]-d;
YklNew[i+4]:=Y_kl[i];
end;
XklNew[30]:=X_kl[29]-d/4;
YklNew[30]:=Y_kl[29]-h;
XklNew[31]:=X_kl[29]-d/2;
YklNew[31]:=Y_kl[29];
XklNew[32]:=X_kl[29]-3*d/4;
YklNew[32]:=Y_kl[29]-h;

XklNew[63]:=XklNew[0];
YklNew[63]:=YklNew[0];
NklNew:=64;

End;

```

```

Procedure Splayn_BR(n1,n2,l:integer; var x,y,xr,yr:array of real; var m:integer);
  Var i,j,k:integer;
    a0,a1,a2,a3,b0,b1,b2,b3,d,t:real;
    xx1,xx2,yy1,yy2,xn,yn,xk,yk:real;
    xx,yy:array[0..100] of real;
  begin
    For i:=n1 to n2 do
      begin
        Xx[i]:=x[i];
        yy[i]:=y[i];
      end;

      m:=-1;
      yy1:=y[n1];
      xx1:=x[n1];
      xx2:=x[n2];
      yy2:=y[n2];
      yy[n1-1]:=yy1;
      xx[n1-1]:=xx1;
      yy[n2+1]:=yy2;
      xx[n2+1]:=xx2;
    end;
  end;

```

```

    m:=m+1;
    xr[m]:=xx1;
    yr[m]:=yy1;

for i:=n1 to n2-1 do
begin

    a3:=(-xx[i-1]+3*(xx[i]-xx[i+1])+xx[i+2])/6;
    a2:=(xx[i-1]-2*xx[i]+xx[i+1])/2;
    a1:=(xx[i+1]-xx[i-1])/2;
    a0:=(xx[i-1]+4*xx[i]+xx[i+1])/6;

    b3:=(-yy[i-1]+3*(yy[i]-yy[i+1])+yy[i+2])/6;
    b2:=(yy[i-1]-2*yy[i]+yy[i+1])/2;
    b1:=(yy[i+1]-yy[i-1])/2;
    b0:=(yy[i-1]+4*yy[i]+yy[i+1])/6;

    k:=1;

    for j:=m+1 to m+l do
    begin
        t:= k/l;
        k:=k+1;
        xr[j]:=((a3*t+a2)*t+a1)*t+a0;
        yr[j]:=((b3*t+b2)*t+b1)*t+b0;
        { writeln(j:5,xr[j]:6:2,yr[j]:6:2);
          readln}
    end;

    m:=m+L;
end;
m:=m+1;
xr[m]:=xx2;
yr[m]:=yy2;
end;

```

```

procedure TForm1.Button2Click(Sender: TObject);
var
    Xd,Yd,xr,yr:array[0..100]of real;
    Mash_X,Mash_Y,XcE,Xcf,YcE,Ycf,DovgKl,VusotKl,R,Max_X,Min_X,Max_Y,Min_Y,

```

AlfaE,AlfaB,R0,Xa1,Ya1,Xa2,Ya2,Xb1,Yb1,Xb2,Yb2,XF1,YF1,XF2,YF2,Teta1,Xa1r,Ya1r,Xa2r,Ya2r,Xb1r,Yb1r,Xb2r,Yb2r,R_gol,h,A_Pole,B_Pole,Hs:real;
i,n_d,m,m1,n_pol:integer;

Procedure Shlapa1;

var

i:integer;

begin

if PrEdit1 and PrEdit2 and PrEdit3 and PrEdit4 and Pr_Im2 Then

Begin

if (GlGolV>GlGolL) Then

GlGolR:=GlGolV

Else

GlGolR:=GlGolL;

H:=(GlGolR/2+deltaR;

Xo:=0;

Yo:=0;

*alfa:=2*Pi/n;*

R:=(GlGolR/2;

*DovR:=2*Pi*R/n;*

DovKlN:=(ObhGol/n+delta+delta1;

*alfa1:=(alfa*DovKlN/DovR;*

Teta:=(Pi/2-alfa/2;

Teta1:=(Pi/2-alfa1/2;

*Xd1:=(R+deltaR)*cos(-Teta-alfa);*

*Yd1:=(R+deltaR)*sin(-Teta-alfa);*

*XF1:=(R+deltaR)*cos(-Teta1-alfa1);*

*YF1:=(R+deltaR)*sin(-Teta1-alfa1);*

*Xc1:=R/2*cos(-Teta-alfa);*

*Yc1:=R/2*sin(-Teta-alfa);*

*{Xc1:=R/2*cos(-Teta);*

*Yc1:=R/2*sin(-Teta);}*

//Xa1:=(Xc1+Xd1)/2;

//Ya1:=(Yc1+Yd1)/2;

Xb1:=(Xc1+Xo)/2;

Yb1:=(Yc1+Yo)/2;

*Xc2:=R/2*cos(-Teta);*


```

Yc2:=R/2*sin(-Teta);
Xd2:=(R+deltaR)*cos(-Teta);
Yd2:=(R+deltaR)*sin(-Teta);
XF2:=(R+deltaR)*cos(-Teta1);
YF2:=(R+deltaR)*sin(-Teta1);
//Xa2:=(Xc2+Xd2)/2;
//Ya2:=(Yc2+Yd2)/2;
Xb2:=(Xc2+Xo)/2;
Yb2:=(Yc2+Yo)/2;
i:=-1;
I:=i+1; X_kl[i]:=Xd2; Y_kl[i]:=Yd2;
//I:=i+1; X_kl[i]:=Xa2; Y_kl[i]:=Ya2;
I:=i+1; X_KL[i]:=Xc2; Y_KL[i]:=Yc2;
I:=i+1; X_KL[i]:=Xb2; Y_KL[i]:=Yb2;
I:=i+1; X_KL[i]:=Xo; Y_KL[i]:=Yo;
I:=i+1; X_KL[i]:=Xb1; Y_KL[i]:=Yb1;
I:=i+1; X_KL[i]:=Xc1; Y_KL[i]:=Yc1;
//I:=i+1; X_kl[i]:=Xa1; Y_kl[i]:=Ya1;
I:=i+1; X_KL[i]:=Xd1; Y_KL[i]:=Yd1;
I:=i+1; X_KL[i]:=Xd2; Y_KL[i]:=Yd2;
n_kl:=i+1;
Xa1:=(Xc1+Xf1)/2;
Ya1:=(Yc1+Yf1)/2;
Xa2:=(Xc2+Xf2)/2;
Ya2:=(Yc2+Yf2)/2;

i:=-1;
I:=i+1; X_kl0[i]:=XF2; Y_kl0[i]:=YF2;
I:=i+1; X_kl0[i]:=Xa2; Y_kl0[i]:=Ya2;
I:=i+1; X_KL0[i]:=Xc2; Y_KL0[i]:=Yc2;
I:=i+1; X_KL0[i]:=Xb2; Y_KL0[i]:=Yb2;
I:=i+1; X_KL0[i]:=Xo; Y_KL0[i]:=Yo;
I:=i+1; X_KL0[i]:=Xb1; Y_KL0[i]:=Yb1;
I:=i+1; X_KL0[i]:=Xc1; Y_KL0[i]:=Yc1;
I:=i+1; X_KL0[i]:=Xa1; Y_KL0[i]:=Ya1;
I:=i+1; X_KL0[i]:=XF1; Y_KL0[i]:=YF1;
I:=i+1; X_KL0[i]:=XF2; Y_KL0[i]:=YF2;

n_kl0:=i+1;

Max_X:=Max_Min(n_kl,1,X_kl);
Max_Y:=Max_Min(n_kl,1,Y_kl);
Min_X:=Max_Min(n_kl,-1,X_kl);
Min_Y:=Max_Min(n_kl,-1,Y_kl);
DovgKl:=Max_X-Min_X;

```

```

VusotKl:=R+deltaR;
Xcf:=(Max_X+Min_X)/2;
Ycf:=(Max_Y+Min_Y)/2;
XcE:=Form1.Image5.Width/2;
YcE:=Form1.Image5.Height/2;
Mash_X:=(Form1.Image5.Width-50)/DovgKl;
Mash_Y:=(Form1.Image5.Height-50)/VusotKl;
if Mash_X>Mash_Y then MXY_im5:=Mash_Y
else
    MXY_im5:=Mash_X;

AlfaE:=-Teta;
AlfaB:=-Teta-alfa;
R0:=R+DeltaR;

//Graf_im5(n_kl,x_kl,y_kl,MXY_im5,Xcf,Ycf,XcE,YcE,1);
    Duga(Xo,Yo,R0,AlfaB,AlfaE,n_d,Xd,Yd);
//Graf_im5(n_d,xd,yd,MXY_im5,Xcf,Ycf,XcE,YcE,1);
for i:=0 to n_d-1 do
begin
    X_KL[i+n_kl-1]:=Xd[i];
    Y_KL[i+n_kl-1]:=Yd[i];
end;
    n_kl:=n_kl+n_d-1;
    Graf_im5(n_kl,x_kl,y_kl,2/3*MXY_im5,Xcf,Ycf,1/2*XcE,YcE,1,0,1);
//Graf_im5(n_kl0,x_kl0,y_kl0,2/3*MXY_im5,Xcf,Ycf,XcE,YcE,2,1,2);
    AlfaE:=-Teta1;
    AlfaB:=-Teta1-alfa1;
Duga(Xo,Yo,R0,AlfaB,AlfaE,n_d,Xd,Yd);
For i:=0 to n_d-1 do
Begin
    x_kl0[n_kl0-1+i]:=xd[i];
y_kl0[n_kl0-1+i]:=yd[i];
end;
    n_kl0:=n_kl0+n_d-1;
    x_kl0[n_kl0-1]:=x_kl0[0];
    y_kl0[n_kl0-1]:=y_kl0[0];
//Graf_im5(n_d,xd,yd,2/3*MXY_im5,Xcf,Ycf,XcE,YcE,1,0,2);
//Graf_im5(n_kl0,x_kl0,y_kl0,2/3*MXY_im5,Xcf,Ycf,XcE,YcE,1,1,2);
Perpend(x_kl0[0],y_kl0[0],x_kl0[1],y_kl0[1],R0/20, Xa1r,Ya1r,Xa2r,Ya2r);
if xa1r>xa2r then
begin
xa2:=xa1r;
ya2:=ya1r;
end

```

```

else
begin
  xa2:=xa2r;
  ya2:=ya2r;
end;
  Elips(xa2,ya2, 2/3*MXY_im5,Xcf,Ycf,1/2*XcE,YcE);
//Elips(xa1r,ya1r, 2/3*MXY_im5,Xcf,Ycf,XcE,YcE);
//Elips(xa2r,ya2r, 2/3*MXY_im5,Xcf,Ycf,XcE,YcE);
Perpend(x_kl0[2],y_kl0[2],x_kl0[3],y_kl0[3],R0/40, Xb1r,Yb1r,Xb2r,Yb2r);
if xb1r>xb2r then
begin
  xb2:=xb1r;
  yb2:=yb1r;
end
else
begin
  xb2:=xb2r;
  yb2:=yb2r;
end;
  Elips(xb2,yb2, 2/3*MXY_im5,Xcf,Ycf,1/2*XcE,YcE);
//Elips(xb1r,yb1r, 2/3*MXY_im5,Xcf,Ycf,XcE,YcE);
//Elips(xb2r,yb2r, 2/3*MXY_im5,Xcf,Ycf,XcE,YcE);
Perpend(x_kl0[4],y_kl0[4],x_kl0[5],y_kl0[5],R0/40, Xb1r,Yb1r,Xb2r,Yb2r);
if xb1r<xb2r then
begin
  xb1:=xb1r;
  yb1:=yb1r;
end
else
begin
  xb1:=xb2r;
  yb1:=yb2r;
end;
//Elips(xb1r,yb1r, 2/3*MXY_im5,Xcf,Ycf,XcE,YcE);
//Elips(xb2r,yb2r, 2/3*MXY_im5,Xcf,Ycf,XcE,YcE);
  Elips(xb2,yb2, 2/3*MXY_im5,Xcf,Ycf,1/2*XcE,YcE);
  Perpend(x_kl0[6],y_kl0[6],x_kl0[7],y_kl0[7],R0/20, Xa1r,Ya1r,Xa2r,Ya2r);
if xa1r<xa2r then
begin
  xa1:=xa1r;
  ya1:=ya1r;
end
else
begin
  xa1:=xa2r;

```

```

    ya1:=ya2r;
end;
    Elips(xa1,ya1, 2/3*MXY_im5,Xcf,Ycf,1/2*XcE,YcE);

//Elips(xa1r,ya1r, 2/3*MXY_im5,Xcf,Ycf,XcE,YcE);
//Elips(xa2r,ya2r, 2/3*MXY_im5,Xcf,Ycf,XcE,YcE);
    x_kl0[1]:=xa2;
    y_kl0[1]:=ya2;
    x_kl0[3]:=xb2;
    y_kl0[3]:=yb2;
    x_kl0[5]:=xb1;
    y_kl0[5]:=yb1;
    x_kl0[7]:=xa1;
    y_kl0[7]:=ya1;
//Graf_im5(n_kl0,x_kl0,y_kl0,2/3*MXY_im5,Xcf,Ycf,XcE,YcE,2,1,2);
    Splayn_BR(0,4,7,x_kl0,y_kl0,xr,yr,m);
//Graf_im5(m,xr,yr,2/3*MXY_im5,Xcf,Ycf,XcE,YcE,2,1,2);
    for i:=n_kl0-1 downto 5 do
    begin
        x_kl0[i+m-4]:=x_kl0[i];
        y_kl0[i+m-4]:=y_kl0[i];
    end;
    for i:=0 to m do
    begin
        x_kl0[i]:=xr[i];
        y_kl0[i]:=yr[i];
    end;
    n_kl0:=n_kl0+m-4;
    Splayn_BR(m,m+4,7,x_kl0,y_kl0,xr,yr,m1);
//Graf_im5(n_kl0,x_kl0,y_kl0,2/3*MXY_im5,Xcf,Ycf,XcE,YcE,2,1,2);
//Graf_im5(m1,xr,yr,2/3*MXY_im5,Xcf,Ycf,XcE,YcE,2,1,2);
    for i:=n_kl0-1 downto m do
    begin
        x_kl0[i+m1-4]:=x_kl0[i];
        y_kl0[i+m1-4]:=y_kl0[i];
    end;
    for i:=m to m+m1 do
    begin
        x_kl0[i]:=xr[i-m];
        y_kl0[i]:=yr[i-m];
    end;
    n_kl0:=n_kl0+m1-4;
    Graf_im5(n_kl0,x_kl0,y_kl0,2/3*MXY_im5,Xcf,Ycf,1/2*XcE,YcE,2,1,2);
    R_gol:=ObhGol/2/pi;
    h:=100;

```

```

for i:=0 to 18 do
begin
  X_pol[i]:=R_gol*cos(-pi*i/18);
  Y_pol[i]:=R_gol*sin(-pi*i/18);
end;
for i:=0 to 18 do
begin
  X_pol[i+19]:=(R_gol+h)*cos(pi+pi*i/18);
  Y_pol[i+19]:=(R_gol+h)*sin(pi+pi*i/18);
end;
X_pol[38]:=X_pol[0];
Y_pol[38]:=Y_pol[0];
n_pol:=39;
Graf_im5(n_pol,x_pol,y_pol,1/3*MXY_im5,Xcf,Ycf,3/2*XcE,YcE,2,1,2);

```

```

model:='Жіноча шляпа з широкими полями';
col_det:=2;
name_det[1]:='Клин шляпи';
name_det[2]:='Поле шляпи';
col_point_det[1]:=n_kl0;
col_point_det[2]:=n_pol;
for i:=0 to col_point_det[1]-1 do
begin
  x_det[1,i]:=x_kl0[i]; y_det[1,i]:=y_kl0[i];
end;
for i:=0 to col_point_det[2]-1 do
begin
  x_det[2,i]:=x_pol[i]; y_det[2,i]:=y_pol[i];
end;
//x_det[2,0]:=x_det[2,col_point_det[2]-1];
//y_det[2,0]:=y_det[2,col_point_det[2]-1];

```

```

With Image5.Canvas do
begin

```

```

  Font.Name:='Ariel';
  Font.Size:=16;
  Font.Color:=clBlack;

```

```

  TextOut(round(Form1.Image1.Width/18), round(Form1.Image1.Height/18), 'Жіноча
  пананама з широкими полями');

```

```

  Font.Name:='Ariel';
  Font.Size:=14;
  Font.Color:=clGray;

```

```

//TextOut(round(Form1.Image1.Width/2-50), round(Form1.Image1.Height/4-30),
'1.Основа панаму');
//TextOut(round(Form1.Image1.Width/2-50), round(Form1.Image1.Height/4+130),
'2.Поле панаму');
TextOut(round(Form1.Image1.Width/3-130), round(Form1.Image1.Height/2+130),
'1.Клин шляпу');
TextOut(round(Form1.Image1.Width*2/3+20), round(Form1.Image1.Height/2),
'2.Поле шляпу');
end;

end;
end;

// Помилка!! Неправильно виводить, масштаб
Procedure Shlapa2; //!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!
var
i:integer;
t,l,Fi,H:real;
xr1,yr1,xr2,yr2,xr3,yr3:array[0..100] of real;

begin

if PrEdit1 and PrEdit2 and PrEdit3 and PrEdit4 and Pr_Im3 Then
begin
t:=GlGolV-264;
DugaElips(0,0,17,92,0,pi/2,18,xr1,yr1);
DugaElips(0,-35,93,127,pi/2,pi,18,xr2,yr2);
DugaElips(0,-35,93,5,pi,3*pi/2,9,xr3,yr3);

for i:=0 to 18 do
begin
x_kl2[i]:=xr1[i];
y_kl2[i]:=yr1[i];
end;

for i:=1 to 18 do
begin
x_kl2[18+i]:=xr2[i];
y_kl2[18+i]:=yr2[i];
end;

for i:=1 to 9 do
begin
x_kl2[36+i]:=xr3[i];
y_kl2[36+i]:=yr3[i];

```

end;

x_kl2[46]:=17;
y_kl2[46]:=-40;

x_kl2[47]:=x_kl2[0];
y_kl2[47]:=y_kl2[0];

n_kl2:=48;

Max_X:=Max_Min(n_kl2,1,X_kl2);
Max_Y:=Max_Min(n_kl2,1,Y_kl2);
Min_X:=Max_Min(n_kl2,-1,X_kl2);
Min_Y:=Max_Min(n_kl2,-1,Y_kl2);
DovgKl:=Max_X-Min_X;
VusotKl:=Max_Y-Min_Y;
Xcf:=(Max_X+Min_X)/2;
Ycf:=(Max_Y+Min_Y)/2;
XcE:=Form1.Image5.Width/2;
YcE:=Form1.Image5.Height/2;
Mash_X:=(Form1.Image5.Width-50)/DovgKl;
Mash_Y:=(Form1.Image5.Height-50)/VusotKl;
if Mash_X>Mash_Y then MXY_im5:=Mash_Y
else
 MXY_im5:=Mash_X;

*Graf_im5(n_kl2,x_kl2,y_kl2,1/3*MXY_im5,Xcf,Ycf,1/3*XcE+30,3/2*YcE,1,0,1);*

for i:=0 to n_kl2 do

begin

x_kl2z[i]:=-x_kl2[i];

y_kl2z[i]:=y_kl2[i];

end;

*Graf_im5(n_kl2,x_kl2z,y_kl2z,1/3*MXY_im5,Xcf,Ycf,2/3*XcE+250,3/2*YcE,1,0,1);*

*DugaElips(0,t/2,175,7,pi,2*pi,18,xr1,yr1);*

DugaElips(0,-t/2,175,7,0,pi,18,xr2,yr2);

for i:=0 to 18 do

begin

x_osn[i]:=xr1[i];

y_osn[i]:=yr1[i];

end;

```

for i:=0 to 18 do
begin
x_osn[19+i]:=xr2[i];
y_osn[19+i]:=yr2[i];
end;

x_osn[38]:=x_osn[0];
y_osn[38]:=y_osn[0];
n_osn:=39;

Max_X:=Max_Min(n_osn,1,X_osn);
Max_Y:=Max_Min(n_osn,1,Y_osn);
Min_X:=Max_Min(n_osn,-1,X_osn);
Min_Y:=Max_Min(n_osn,-1,Y_osn);
Xcf:=(Max_X+Min_X)/2;
Ycf:=(Max_Y+Min_Y)/2;

Graf_im5(n_osn,x_osn,y_osn,1/3*MXY_im5,Xcf,Ycf,XcE,1/3*YcE,1,0,1);
Pole(pi/2,ObhGol,l,Fi);
GenPole(L,50,Fi,Npol, Xpol,Ypol);
Graf_im5(Npol,Xpol,Ypol,1/3*MXY_im5,Xcf,Ycf,XcE,1/3*YcE,1,0,1);

model:='Жіноча панама з неширокими полями';
col_det:=4;
name_det[1]:='Клин (1)';
name_det[2]:='Клин (2)';
name_det[3]:='Основа панамы';
name_det[4]:='Поле панамы';
col_point_det[1]:=n_kl2;
col_point_det[2]:=n_kl2;
col_point_det[3]:=n_osn;
col_point_det[4]:=Npol;
for i:=0 to col_point_det[1]-1 do
begin
x_det[1,i]:=x_kl2[i]; y_det[1,i]:=y_kl2[i];
end;
for i:=0 to col_point_det[2]-1 do
begin
x_det[2,i]:=x_kl2z[i]; y_det[2,i]:=y_kl2z[i];
end;
for i:=0 to col_point_det[3]-1 do
begin
x_det[3,i]:=x_osn[i]; y_det[3,i]:=y_osn[i];
end;

```



```

for i:=0 to col_point_det[4]-1 do
begin
x_det[4,i]:=Xpol[i]; y_det[4,i]:=Ypol[i];
end;

```

```

With Image5.Canvas do
begin

```

```

Font.Name:='Ariel';
Font.Size:=16;
Font.Color:=clBlack;

```

```

TextOut(round(Form1.Image1.Width/18-20), round(Form1.Image1.Height/18-
30), 'Жіноча панамы з неширокімі полямі');

```

```

Font.Name:='Ariel';
Font.Size:=14;
Font.Color:=clGray;
TextOut(round(Form1.Image1.Width/2-50), round(Form1.Image1.Height/4-30),
'1.Основа панамы');
TextOut(round(Form1.Image1.Width/2-50), round(Form1.Image1.Height/4+130),
'2.Поле панамы');
TextOut(round(Form1.Image1.Width/3-130), round(Form1.Image1.Height/2+130),
'3.Клин (1)');
TextOut(round(Form1.Image1.Width*2/3+50),
round(Form1.Image1.Height/2+130), '4.Клин (2)');
end;
end;
end;

```

```

// Помилка!! Неправильно виводить, масштаб
//!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!

```

```

Procedure Shlapa3;

```

```

var

```

```

i:integer;

```

```

Procedure Okil(a,b:real; var n:integer; var x,y:array of real);

```

```

begin
x[0]:=a/2;
y[0]:=b/2;
x[1]:=-a/2;
y[1]:=b/2;
x[2]:=-a/2;
y[2]:=-b/2;
x[3]:=a/2;

```

```

y[3]:=-b/2;
x[4]:=a/2;
y[4]:=b/2;
n:=5;
end;

```

```

begin
if PrEdit1 and PrEdit2 and PrEdit3 and PrEdit4 and Pr_Im6 Then

```

```

Begin

```

```

n_kl:=0;
n_kl0:=0;

```

```

if (GlGolV>GlGolL) Then
  GlGolR:=GlGolV

```

```

Else

```

```

  GlGolR:=GlGolL;
  H:=GlGolR/2+deltaR;
  Xo:=0;
  Yo:=0;
  alfa:=2*Pi/n;
  R:=GlGolR/2;
  DovR:=2*Pi*R/n;
  DovKlN:=ObhGol/n+delta+delta1;
  alfa1:=alfa*DovKlN/DovR;
  Teta:=Pi/2-alfa/2;
  Teta1:=Pi/2-alfa1/2;
  Xd1:=(R+deltaR)*cos(-Teta-alfa);
  Yd1:=(R+deltaR)*sin(-Teta-alfa);
  XF1:=(R+deltaR)*cos(-Teta1-alfa1);
  YF1:=(R+deltaR)*sin(-Teta1-alfa1);

```

```

  Xc1:=R/2*cos(-Teta-alfa);
  Yc1:=R/2*sin(-Teta-alfa);

```

```

{Xc1:=R/2*cos(-Teta);
Yc1:=R/2*sin(-Teta);}
//Xa1:=(Xc1+Xd1)/2;
//Ya1:=(Yc1+Yd1)/2;
  Xb1:=(Xc1+Xo)/2;
  Yb1:=(Yc1+Yo)/2;
  Xc2:=R/2*cos(-Teta);
  Yc2:=R/2*sin(-Teta);
  Xd2:=(R+deltaR)*cos(-Teta);
  Yd2:=(R+deltaR)*sin(-Teta);

```

```

XF2:=(R+deltaR)*cos(-Teta1);
YF2:=(R+deltaR)*sin(-Teta1);
//Xa2:=(Xc2+Xd2)/2;
//Ya2:=(Yc2+Yd2)/2;
Xb2:=(Xc2+Xo)/2;
Yb2:=(Yc2+Yo)/2;
i:=-1;
I:=i+1; X_kl[i]:=Xd2; Y_kl[i]:=Yd2;
//I:=i+1; X_kl[i]:=Xa2; Y_kl[i]:=Ya2;
I:=i+1; X_KL[i]:=Xc2; Y_KL[i]:=Yc2;
I:=i+1; X_KL[i]:=Xb2; Y_KL[i]:=Yb2;
I:=i+1; X_KL[i]:=Xo; Y_KL[i]:=Yo;
I:=i+1; X_KL[i]:=Xb1; Y_KL[i]:=Yb1;
I:=i+1; X_KL[i]:=Xc1; Y_KL[i]:=Yc1;
//I:=i+1; X_kl[i]:=Xa1; Y_kl[i]:=Ya1;
I:=i+1; X_KL[i]:=Xd1; Y_KL[i]:=Yd1;
I:=i+1; X_KL[i]:=Xd2; Y_KL[i]:=Yd2;
n_kl:=i+1;
Xa1:=(Xc1+Xf1)/2;
Ya1:=(Yc1+Yf1)/2;
Xa2:=(Xc2+Xf2)/2;
Ya2:=(Yc2+Yf2)/2;

i:=-1;
I:=i+1; X_kl0[i]:=XF2; Y_kl0[i]:=YF2;
I:=i+1; X_kl0[i]:=Xa2; Y_kl0[i]:=Ya2;
I:=i+1; X_KL0[i]:=Xc2; Y_KL0[i]:=Yc2;
I:=i+1; X_KL0[i]:=Xb2; Y_KL0[i]:=Yb2;
I:=i+1; X_KL0[i]:=Xo; Y_KL0[i]:=Yo;
I:=i+1; X_KL0[i]:=Xb1; Y_KL0[i]:=Yb1;
I:=i+1; X_KL0[i]:=Xc1; Y_KL0[i]:=Yc1;
I:=i+1; X_KL0[i]:=Xa1; Y_KL0[i]:=Ya1;
I:=i+1; X_KL0[i]:=XF1; Y_KL0[i]:=YF1;
I:=i+1; X_KL0[i]:=XF2; Y_KL0[i]:=YF2;

n_kl0:=i+1;

Max_X:=Max_Min(n_kl,1,X_kl);
Max_Y:=Max_Min(n_kl,1,Y_kl);
Min_X:=Max_Min(n_kl,-1,X_kl);
Min_Y:=Max_Min(n_kl,-1,Y_kl);
DovgKl:=Max_X-Min_X;
VusotKl:=R+deltaR;
Xcf:=(Max_X+Min_X)/2;
Ycf:=(Max_Y+Min_Y)/2;

```

```

XcE:=Form1.Image5.Width/2;
YcE:=Form1.Image5.Height/2;
Mash_X:=(Form1.Image5.Width-50)/DovgKl;
Mash_Y:=(Form1.Image5.Height-50)/VusotKl;
if Mash_X>Mash_Y then MXY_im5:=Mash_Y
else
    MXY_im5:=Mash_X;

    AlfaE:=-Teta;
    AlfaB:=-Teta-alfa;
    R0:=R+DeltaR;

//Graf_im5(n_kl,x_kl,y_kl,MXY_im5,Xcf,Ycf,XcE,YcE,1);
    Duga(Xo,Yo,R0,AlfaB,AlfaE,n_d,Xd,Yd);
//Graf_im5(n_d,xd,yd,MXY_im5,Xcf,Ycf,XcE,YcE,1);
for i:=0 to n_d-1 do
begin
    X_KL[i+n_kl-1]:=Xd[i];
    Y_KL[i+n_kl-1]:=Yd[i];
end;
    n_kl:=n_kl+n_d-1;

//Graf_im5(n_kl,x_kl,y_kl,1/3*MXY_im5,Xcf,Ycf,1/2*XcE,YcE,1,0,1);

//Graf_im5(n_kl0,x_kl0,y_kl0,2/3*MXY_im5,Xcf,Ycf,XcE,YcE,2,1,2);
    AlfaE:=-Teta1;
    AlfaB:=-Teta1-alfa1;
    Duga(Xo,Yo,R0,AlfaB,AlfaE,n_d,Xd,Yd);
For i:=0 to n_d-1 do
Begin
    x_kl0[n_kl0-1+i]:=xd[i];
    y_kl0[n_kl0-1+i]:=yd[i];
end;
    n_kl0:=n_kl0+n_d-1;
    x_kl0[n_kl0-1]:=x_kl0[0];
    y_kl0[n_kl0-1]:=y_kl0[0];

```

```

//Graf_im5(n_d,xd,yd,2/3*MXY_im5,Xcf,Ycf,XcE,YcE,1,0,2);
//Graf_im5(n_kl0,x_kl0,y_kl0,2/3*MXY_im5,Xcf,Ycf,XcE,YcE,1,1,2);
Perpend(x_kl0[0],y_kl0[0],x_kl0[1],y_kl0[1],R0/20, Xa1r,Ya1r,Xa2r,Ya2r);
if xa1r>xa2r then
begin
xa2:=xa1r;
ya2:=ya1r;
end
else
begin
xa2:=xa2r;
ya2:=ya2r;
end;
Elips(xa2,ya2, 2/3*MXY_im5,Xcf,Ycf,1/2*XcE,YcE);
//Elips(xa1r,ya1r, 2/3*MXY_im5,Xcf,Ycf,XcE,YcE);
//Elips(xa2r,ya2r, 2/3*MXY_im5,Xcf,Ycf,XcE,YcE);
Perpend(x_kl0[2],y_kl0[2],x_kl0[3],y_kl0[3],R0/40, Xb1r,Yb1r,Xb2r,Yb2r);
if xb1r>xb2r then
begin
xb2:=xb1r;
yb2:=yb1r;
end
else
begin
xb2:=xb2r;
yb2:=yb2r;
end;
Elips(xb2,yb2, 2/3*MXY_im5,Xcf,Ycf,1/2*XcE,YcE);
//Elips(xb1r,yb1r, 2/3*MXY_im5,Xcf,Ycf,XcE,YcE);
//Elips(xb2r,yb2r, 2/3*MXY_im5,Xcf,Ycf,XcE,YcE);
Perpend(x_kl0[4],y_kl0[4],x_kl0[5],y_kl0[5],R0/40, Xb1r,Yb1r,Xb2r,Yb2r);
if xb1r<xb2r then
begin
xb1:=xb1r;
yb1:=yb1r;
end
else
begin
xb1:=xb2r;
yb1:=yb2r;
end;
//Elips(xb1r,yb1r, 2/3*MXY_im5,Xcf,Ycf,XcE,YcE);
//Elips(xb2r,yb2r, 2/3*MXY_im5,Xcf,Ycf,XcE,YcE);
Elips(xb2,yb2, 2/3*MXY_im5,Xcf,Ycf,1/2*XcE,YcE);
Perpend(x_kl0[6],y_kl0[6],x_kl0[7],y_kl0[7],R0/20, Xa1r,Ya1r,Xa2r,Ya2r);

```

```

if xa1r<xa2r then
begin
  xa1:=xa1r;
ya1:=ya1r;
end
else
begin
  xa1:=xa2r;
ya1:=ya2r;
end;
  Elips(xa1,ya1, 2/3*MXY_im5,Xcf,Ycf,1/2*XcE,YcE);

//Elips(xa1r,ya1r, 2/3*MXY_im5,Xcf,Ycf,XcE,YcE);
//Elips(xa2r,ya2r, 2/3*MXY_im5,Xcf,Ycf,XcE,YcE);
  x_kl0[1]:=xa2;
  y_kl0[1]:=ya2;
  x_kl0[3]:=xb2;
  y_kl0[3]:=yb2;
  x_kl0[5]:=xb1;
  y_kl0[5]:=yb1;
  x_kl0[7]:=xa1;
  y_kl0[7]:=ya1;
//Graf_im5(n_kl0,x_kl0,y_kl0,2/3*MXY_im5,Xcf,Ycf,XcE,YcE,2,1,2);
  Splayn_BR(0,4,7,x_kl0,y_kl0,xr,yr,m);
//Graf_im5(m,xr,yr,2/3*MXY_im5,Xcf,Ycf,XcE,YcE,2,1,2);
  for i:=n_kl0-1 downto 5 do
  begin
    x_kl0[i+m-4]:=x_kl0[i];
    y_kl0[i+m-4]:=y_kl0[i];
  end;
  for i:=0 to m do
  begin
    x_kl0[i]:=xr[i];
    y_kl0[i]:=yr[i];
  end;
  n_kl0:=n_kl0+m-4;
  Splayn_BR(m,m+4,7,x_kl0,y_kl0,xr,yr,m1);
//Graf_im5(n_kl0,x_kl0,y_kl0,2/3*MXY_im5,Xcf,Ycf,XcE,YcE,2,1,2);
//Graf_im5(m1,xr,yr,2/3*MXY_im5,Xcf,Ycf,XcE,YcE,2,1,2);
  for i:=n_kl0-1 downto m do
  begin
    x_kl0[i+m1-4]:=x_kl0[i];
    y_kl0[i+m1-4]:=y_kl0[i];
  end;
  for i:=m to m+m1 do

```

```

begin
  x_kl0[i]:=xr[i-m];
  y_kl0[i]:=yr[i-m];
end;
n_kl0:=n_kl0+m1-4;
n_kl0:=61;
x_kl0[60]:=x_kl0[0];
y_kl0[60]:=y_kl0[0];
  Graf_im5(n_kl0,x_kl0,y_kl0,1/3*MXY_im5,Xcf,Ycf,1/2*XcE,YcE+100,2,1,2);

  PeretvorKluna(n_kl0, x_kl0,y_kl0,30, NklNew, XklNew,YklNew);
  Graf_im5(NklNew,XklNew,YklNew,1/3*MXY_im5,Xcf,Ycf,5/3*XcE-
50,YcE+100,2,1,2);

  Xcf:=0;
  Ycf:=0;
  Okil(ObhGol,100,col_point_det[3],x_det[3],y_det[3]);
  Graf_im5(col_point_det[3],x_det[3],y_det[3],1/3*MXY_im5,Xcf,Ycf,XcE,YcE-
100,1,0,1);
  model:='Жіноча панاما з шнуровкою';
  col_det:=3;
  name_det[1]:='Клин панамми';
  name_det[2]:='Клин панамми з шнуровкою';
  name_det[3]:='Окіл';

  col_point_det[1]:=n_kl0;
  col_point_det[2]:=NklNew;

  for i:=0 to col_point_det[1]-1 do
  begin
    x_det[1,i]:=x_kl0[i]; y_det[1,i]:=y_kl0[i];
  end;
  for i:=0 to col_point_det[2]-1 do
  begin
    x_det[2,i]:=XklNew[i]; y_det[2,i]:=YklNew[i];
  end;

  With Image5.Canvas do
  begin

    Font.Name:='Ariel';
    Font.Size:=16;
    Font.Color:=clBlack;

```

TextOut(round(Form1.Image1.Width/18), round(Form1.Image1.Height/18), 'Жіноча пананама з декоративною шнуровкою');

```

Font.Name:='Ariel';
Font.Size:=14;
Font.Color:=clGray;
TextOut(round(Form1.Image1.Width/3-130), round(Form1.Image1.Height*7/10-40), '2.Клин пананами');
TextOut(round(Form1.Image1.Width/2-50), round(Form1.Image1.Height/4+40), '1.Окіл пананами');
TextOut(round(Form1.Image1.Width*2/3-80), round(Form1.Image1.Height/2+110), '3.Клин пананами з шнуровкою');
end;

```

end;

```

End;
{Procedure Shlapa4;
var
i:integer;
t,l,Fi,H:real;
xr1,yr1,xr2,yr2,xr3,yr3:array[0..100] of real;

```

begin

*if PrEdit1 and PrEdit2 and PrEdit3 and PrEdit4 and Pr_Im7 Then
begin*

```

t:=GlGolV-264;
DugaElips(0,0,17,92,0,pi/2,18,xr1,yr1);
DugaElips(0,-35,93,127,pi/2,pi,18,xr2,yr2);
DugaElips(0,-35,93,5,pi,3*pi/2,9,xr3,yr3);

```

for i:=0 to 18 do

begin

```

x_kl2[i]:=xr1[i];
y_kl2[i]:=yr1[i];
end;

```

for i:=1 to 18 do

begin

```

x_kl2[18+i]:=xr2[i];
y_kl2[18+i]:=yr2[i];
end;

```



```

for i:=1 to 9 do
begin
x_kl2[36+i]:=xr3[i];
y_kl2[36+i]:=yr3[i];
end;

x_kl2[46]:=17;
y_kl2[46]:=-40;

x_kl2[47]:=x_kl2[0];
y_kl2[47]:=y_kl2[0];

n_kl2:=48;

Max_X:=Max_Min(n_kl2,1,X_kl2);
Max_Y:=Max_Min(n_kl2,1,Y_kl2);
Min_X:=Max_Min(n_kl2,-1,X_kl2);
Min_Y:=Max_Min(n_kl2,-1,Y_kl2);
DovgKl:=Max_X-Min_X;
VusotKl:=Max_Y-Min_Y;
Xcf:=(Max_X+Min_X)/2;
Ycf:=(Max_Y+Min_Y)/2;
XcE:=Form1.Image5.Width/2;
YcE:=Form1.Image5.Height/2;
Mash_X:=(Form1.Image5.Width-50)/DovgKl;
Mash_Y:=(Form1.Image5.Height-50)/VusotKl;
if Mash_X>Mash_Y then MXY_im5:=Mash_Y
else
MXY_im5:=Mash_X;

Graf_im5(n_kl2,x_kl2,y_kl2,1/3*MXY_im5,Xcf,Ycf,1/3*XcE,3/2*YcE,1,0,1);

for i:=0 to n_kl2 do
begin
x_kl2z[i]:=-x_kl2[i];
y_kl2z[i]:=y_kl2[i];
end;
Graf_im5(n_kl2,x_kl2z,y_kl2z,1/3*MXY_im5,Xcf,Ycf,2/3*XcE,3/2*YcE,1,0,1);

DugaElips(0,t/2,175,7,pi,2*pi,18,xr1,yr1);
DugaElips(0,-t/2,175,7,0,pi,18,xr2,yr2);

for i:=0 to 18 do

```

```

begin
x_osn[i]:=xr1[i];
y_osn[i]:=yr1[i];
end;

for i:=0 to 18 do
begin
x_osn[19+i]:=xr2[i];
y_osn[19+i]:=yr2[i];
end;

x_osn[38]:=x_osn[0];
y_osn[38]:=y_osn[0];
n_osn:=39;

Max_X:=Max_Min(n_osn,1,X_osn);
Max_Y:=Max_Min(n_osn,1,Y_osn);
Min_X:=Max_Min(n_osn,-1,X_osn);
Min_Y:=Max_Min(n_osn,-1,Y_osn);
Xcf:=(Max_X+Min_X)/2;
Ycf:=(Max_Y+Min_Y)/2;

Graf_im5(n_osn,x_osn,y_osn,1/3*MXI_im5,Xcf,Ycf,XcE,1/3*YcE,1,0,1);
Pole(pi/2,ObhGol,l,Fi);
GenPole(L,50,Fi,Npol, Xpol,Ypol);
Graf_im5(Npol,Xpol,Ypol,1/3*MXI_im5,Xcf,Ycf,XcE,1/3*YcE,1,0,1);
end;
end; }

```

Procedure Shlapa5;

```

var
i, n_r:integer;
r,Dovg,Vusota,Xcf,Ycf,XcE,YcE,L,a,b,delta,H:real;
x_r1,y_r1,x_r2,y_r2,x_r3,y_r3,x_r4,y_r4:array[0..50]of real;

begin
model:='Конфідератка';
col_det:=2;
name_det[1]:='Верхня частина конфідератки';
name_det[2]:='Окіл';
if PrEdit1 and PrEdit2 and PrEdit3 and PrEdit4 and Pr_Im8 Then

```

Begin

a:=340;

*b:=1.44*a/2;*

delta:=40;

H:=250;

col_point_det[2]:=5;

L:=ObhGol+delta;

x_det[2,0]:=0; y_det[2,0]:=0;

x_det[2,1]:=L; y_det[2,1]:=0;

x_det[2,2]:=L; y_det[2,2]:=H;

x_det[2,3]:=0; y_det[2,3]:=H;

x_det[2,4]:=0; y_det[2,4]:=0;

r:=2(ObhGol/4+delta)/pi;*

*Duga(a/2,a/2,r,pi,3/2*pi, n_r, x_r1,y_r1);*

Duga(a/2,-a/2,r,pi/2,pi, n_r,x_r2,y_r2);

Duga(-a/2,-a/2,r,0,pi/2, n_r, x_r3,y_r3);

*Duga(-a/2,a/2,r,3/2*pi,2*pi, n_r, x_r4,y_r4);*

n_r:=18;

for i:=0 to n_r do

begin

x_det[1,i]:=x_r1[i];

y_det[1,i]:=y_r1[i];

end;

for i:=0 to n_r do

begin

x_det[1,i+n_r+1]:=x_r2[i];

y_det[1,i+n_r+1]:=y_r2[i];

end;

for i:=0 to n_r do

begin

*x_det[1,i+2*n_r+2]:=x_r3[i];*

*y_det[1,i+2*n_r+2]:=y_r3[i];*

end;

for i:=0 to n_r do

begin

*x_det[1,i+3*n_r+3]:=x_r4[i];*

*y_det[1,i+3*n_r+3]:=y_r4[i];*

end;

*x_det[1,4*n_r+4]:=x_det[1,0];*

```

y_det[1,4*n_r+4]:=y_det[1,0];
col_point_det[1]:=4*n_r+5;

Xcf:=0;
Ycf:=0;

XcE:=Form1.Image5.Width/2;
YcE:=Form1.Image5.Height/2;

Mash_X:=(Form1.Image5.Width-50)/(ObhGol+delta);
Mash_Y:=(Form1.Image5.Height-50)/(a+delta);
if Mash_X>Mash_Y then MXY_im5:=Mash_Y
else
    MXY_im5:=Mash_X;

Graf_im5(col_point_det[1],x_det[1],y_det[1],1/2*MXY_im5,Xcf,Ycf,XcE,1/2*YcE,1,
0,1);

Graf_im5(col_point_det[2],x_det[2],y_det[2],1/2*MXY_im5,Xcf,Ycf,XcE-
150,7/4*YcE,1,0,1);

With Image5.Canvas do
begin

Font.Name:='Ariel';
Font.Size:=16;
Font.Color:=clBlack;

TextOut(round(Form1.Image1.Width/18),
round(Form1.Image1.Height/18),'Конфідератка');

Font.Name:='Ariel';
Font.Size:=14;
Font.Color:=clGray;
TextOut(round(Form1.Image1.Width/3), round(Form1.Image1.Height/3), '1.Верхня
частина конфідератки');
TextOut(round(Form1.Image1.Width/3), round(Form1.Image1.Height*7/10),
'2.Бокова сторона конфідератки');
end;

end;
End;

```

Procedure ShlapaKovboj;