

Міністерство освіти і науки України
Національний університет “Львівська Політехніка”

Лабораторна робота №5А
З дисципліни
“Програмування частина 2”

Виконав:
Студент групи АП-11
Гишка Остап

Прийняв:
Чайковський І.Б.

Львів-2024

Тема роботи: Дослідження циклічних операторів мови програмування C.

Мета роботи: Дослідити властивості циклічних операторів мови C.

Приклад 1

```
#include <stdio.h>
#include <math.h>
#define PI 3.14159265
#define EXP 2.71828182
int main(void) {
    float y;
    int N = 32;
    float a = INFINITY; // Початкове значення для мінімуму
    float b = -INFINITY; // Початкове значення для максимуму
    float res[N]; // Масив для зберігання значень функції
    for (int i = 0; i < N; i++) {
        y = pow(i, 2) * pow(EXP, (-pow(i, 2) / 100.0)) * sin((2 * PI / N) * i);
        res[i] = y;
        b = fmax(b, y); // Використовуємо fmax для визначення максимуму
        a = fmin(a, y); // Використовуємо fmin для визначення мінімуму
        printf("i = %d, y = %f\n", i, y);
    }
    printf("max = %f\n", b);
    printf("min = %f\n", a);
    return 0;}
```

```
i = 1, y = 0.193149
i = 2, y = 1.470713
i = 3, y = 4.569777
i = 4, y = 9.640906
i = 5, y = 16.188730
i = 6, y = 23.204479
i = 7, y = 29.441893
i = 8, y = 33.746716
i = 9, y = 35.341129
i = 10, y = 33.987629
i = 11, y = 30.000978
i = 12, y = 24.124784
i = 13, y = 17.324791
i = 14, y = 10.565220
i = 15, y = 4.626533
i = 16, y = 0.000000
i = 17, y = -3.133448
i = 18, y = -4.855909
i = 19, y = -5.425541
i = 20, y = -5.180445
i = 21, y = -4.457038
i = 22, y = -3.535700
i = 23, y = -2.615844
i = 24, y = -1.815040
i = 25, y = -1.183351
i = 26, y = -0.723988
i = 27, y = -0.413587
i = 28, y = -0.218239
i = 29, y = -0.104020
i = 30, y = -0.042504
i = 31, y = -0.012572
max = 35.341129
min = -5.425541
```

Приклад 2

//FLOAT

```
#include <stdio.h>
#include <math.h>
int main(void) {
    int i = 0;
    float precision = 1.0, a = 1.0 + precision;
    for (precision = 1.0; a > 1.0; ++i) {
        precision = precision / 2;
        a = 1.0 + precision;
    }
    printf("\nЧисло ділень на 2: %6d\n", i);
    printf("Машинний нуль: %e\n", precision);
}
```

```
#include <stdio.h>
#include <math.h>
int main(void) {
    int i = 0;
    float precision = 1.0;
    float a = 1.0 + precision;
    while (a > 1.0) {
        precision = precision / 2;
        a = 1.0 + precision;
        ++i;
    }
    printf("\nЧисло ділень на 2: %6d\n", i);
    printf("Машинний нуль: %e\n", precision);
}
```

```
#include<stdio.h>
#include<math.h>
void main(void){
    int i=0;
    float precision,a;
    precision = 1.0;
    do{
        precision = precision/2;
        a = 1.0 + precision;
        ++i;}
    while(a>1);
    printf("\nчисло ділень на 2: %6d\n",i);
    printf("машинний нуль: %e\n ",precision);
}
```

```
число ділень на 2:      24
машинний нуль: 5.960464e-08
```

//double

```
#include <stdio.h>
#include <math.h>
int main(void) {
    int i = 0;
    double precision = 1.0, a = 1.0 + precision;
    for (precision = 1.0; a > 1.0; ++i) {
        precision = precision / 2;
        a = 1.0 + precision;}
    printf("\nЧисло ділень на 2: %6d\n", i);
    printf("Машинний нуль: %e\n", precision);
}
```

```
#include <stdio.h>
#include <math.h>
int main(void) {
    int i = 0;
    double precision = 1.0;
    double a = 1.0 + precision;
    while (a > 1.0) {
        precision = precision / 2;
        a = 1.0 + precision;
        ++i;}
    printf("\nЧисло ділень на 2: %6d\n", i);
    printf("Машинний нуль: %e\n", precision);
}
```

```
#include <stdio.h>
#include <math.h>
int main(void) {
    int i = 0;
    double precision, a;
    precision = 1.0;
    do {
        precision = precision / 2;
        a = 1.0 + precision;
        ++i;
    } while (a > 1.0);
    printf("\nЧисло ділень на 2: %6d\n", i);
    printf("Машинний нуль: %e\n", precision);
}
```

```
Число ділень на 2:      53
Машинний нуль: 1.110223e-16
```

//long double

```
#include <stdio.h>
#include <math.h>
int main(void) {
    int i = 0;
    long double precision = 1.0L, a = 1.0L + precision;
    for (precision = 1.0L; a > 1.0L; ++i) {
        precision = precision / 2;
        a = 1.0L + precision;
    }
    printf("\nЧисло ділень на 2: %6d\n", i);
    printf("Машинний нуль: %Le\n", precision);
}
```

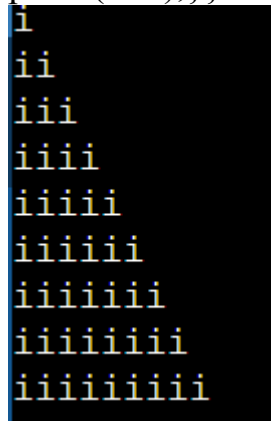
```
#include <stdio.h>
#include <math.h>
int main(void) {
    int i = 0;
    long double precision = 1.0L;
    long double a = 1.0L + precision;
    while (a > 1.0L) {
        precision = precision / 2;
        a = 1.0L + precision;
        ++i;
    }
    printf("\nЧисло ділень на 2: %6d\n", i);
    printf("Машинний нуль: %Le\n", precision);
}
```

```
#include <stdio.h>
#include <math.h>
int main(void) {
    int i = 0;
    long double precision, a;
    precision = 1.0L;
    do {
        precision = precision / 2;
        a = 1.0L + precision;
        ++i;
    } while (a > 1.0L);
    printf("\nЧисло ділень на 2: %6d\n", i);
    printf("Машинний нуль: %Le\n", precision);
}
```

```
Число ділень на 2:      64
Машинний нуль: 5.421011e-20
```

Приклад 3

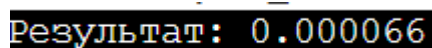
```
#include<stdio.h>
void main(){
char x = 105;
for(int i = 1;i<10; i++){
for(int j = 0;j<i;j++){
printf("%c",x);}
printf("\n");}}
```



```
i
ii
iii
iiii
iiiii
iiiii
iiiii
iiiii
iiiii
iiiii
```

Приклад 4

```
#include <stdio.h>
// Функція для обчислення факторіалу
unsigned long long factorial(int n) {
    unsigned long long fact = 1;
    for (int i = 1; i <= n; i++) {
        fact *= i;
    }
    return fact;
}
int main() {
    int N = 5;
    int M = 5;
    // Обчислення факторіалів чисел M та N
    unsigned long fact_M = factorial(M);
    unsigned long fact_N = factorial(N);
    // Обчислення факторіалу суми M та N
    unsigned long fact_MN = factorial(M + N);
    // Обчислення значення виразу
    double result = (double)(fact_M + fact_N) / fact_MN;
    printf("Результат: %lf\n", result);
    return 0;
}
```



```
Результат: 0.000066
```

Приклад 5

```
#include <stdio.h>
#include <math.h>
#include <windows.h>
float factorial(float n);
float sin_x(float x,float y);
float cos_x(float x,float y);
float exp_x(float x,float y);
int z=0;
void main(){
    SetConsoleCP(65001);
    SetConsoleOutputCP(65001);
    float a = 0.00001;
    float x;
    printf("Введіть значення x в межах  $0 \leq X \leq \pi/2$ :");
    scanf("%f",&x);
    printf("Значення sin(x) за допомогою ітераційного процесу: %f\n", sin_x(x,
a));
    printf("%d\n", z);
    printf("Значення sin(x) за допомогою бібліотечної функції: %f\n\n", sin(x));
    printf("Значення cos(x) за допомогою ітераційного процесу: %f\n", cos_x(x,
a));
    printf("%d\n", z);
    printf("Значення cos(x) за допомогою бібліотечної функції: %f\n\n", cos(x));
    printf("Значення cos(x) за допомогою ітераційного процесу: %f\n", exp_x(x,
a));
    printf("%d\n", z);
    printf("Значення cos(x) за допомогою бібліотечної функції: %f\n", exp(x));}
float factorial(float n){
    if(n==0){
        return 1;}
    else{
        return n*factorial(n-1);}}
float sin_x(float x,float y){
    float n = 0;
    float X = x;
    float sum = x;
    z=0;
    while (fabs(X)>y){
        n = n+1;
        X = pow(-1,n)*(pow(x,2*n+1)/factorial(2*n+1));
        sum = sum + X;
        z = z+1;}
    return sum;}
```

```

float cos_x(float x,float y){
    float n = 0;
    float X = x;
    float sum = 1;
    z=0;
    while (fabs(X)>y){
        n = n+1;
        X = pow(-1,n)*(pow(x,2*n)/factorial(2*n));
        sum = sum + X;
        z = z+1;}
    return sum;}
float exp_x(float x,float y){
    float n = 0;
    float X = x;
    float sum = 1;
    z=0;
    while (fabs(X)>y){
        n = n+1;
        X = (pow(x,n)/factorial(n));
        sum = sum + X;
        z = z+1;}
    return sum;}

```

```

Введіть значення x в межах  $0 \leq X \leq \pi/2$ : 24
Значення sin(x) за допомогою ітераційного процесу: 135743360.000000
17
Значення sin(x) за допомогою бібліотечної функції: -0.905578

Значення cos(x) за допомогою ітераційного процесу: -92049536.000000
18
Значення cos(x) за допомогою бібліотечної функції: 0.424179

Значення cos(x) за допомогою ітераційного процесу: 25944244224.000000
35
Значення cos(x) за допомогою бібліотечної функції: 26489122129.843472

```