

**Національний технічний університет України
“Київський політехнічний інститут”**

Факультет прикладної математики

**Кафедра системного програмування і спеціалізованих
комп'ютерних систем**

ЛАБОРАТОРНА РОБОТА №1.2.

з дисципліни “Структури даних і алгоритми”

**ТЕМА: “АЛГОРИТМИ З ВКЛАДЕНИМИ ЦИКЛАМИ ТА
МЕТОД ДИНАМІЧНОГО ПРОГРАМУВАННЯ”**

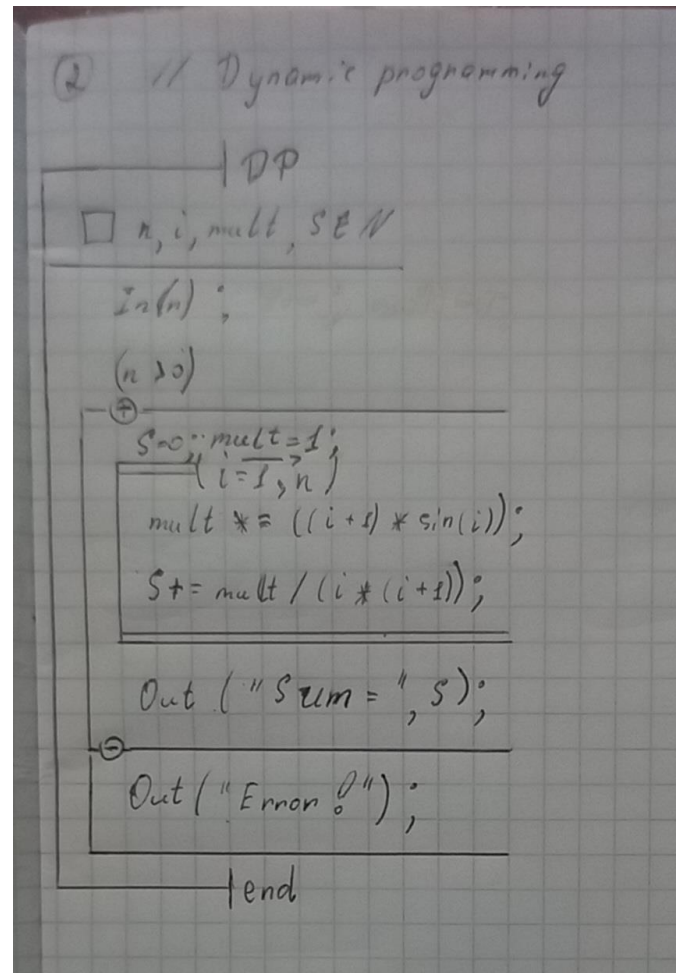
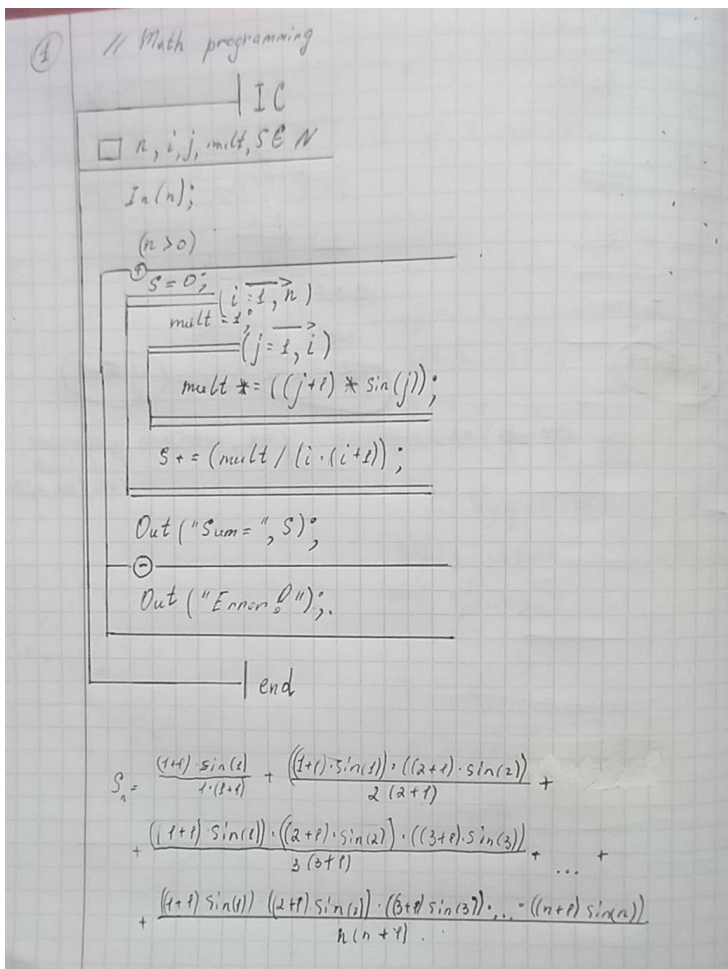
**Група: КВ-34
Виконав: Фесенко Д.О.
Оцінка:**

Постановка задачі

1. Задане натуральне число n . Обчислити значення заданої формули за варіантом.
2. Вирішити поставлену задачу двома способами, написавши дві програми:
 - 1) перша програма повинна використовувати для обчислення формули вкладені цикли;
 - 2) друга програма повинна виконати обчислення формули за допомогою одного циклу з використанням методу динамічного програмування.
3. Виконати розрахунок кількості операцій для кожного з алгоритмів за методикою, викладеною на лекції, додавши до неї **окремий** підрахунок кількості викликів стандартних функцій, позначивши цю кількість KF, а також включити KF до загальної суми кількості операцій.
4. Програма має правильно вирішувати поставлену задачу при будь-якому заданому n , для якого результат обчислення може бути коректно представлений типом **double**.
5. Результати обчислення вивести з достатньою кількістю розрядів дробової частини для коректного показу цих результатів

Завдання за варіантом №23

$$S = \sum_{i=1}^n \frac{\prod_{j=1}^i ((j+1) \cdot \sin(j))}{i \cdot (i+1)}$$



Код програми

```
#include <stdio.h>
#include <math.h>
int main() {
    double mult, S;
    int i, j, n;
    printf("-----|Input your n: ");
    scanf("%d", &n);
    if (n > 0) {
        printf("-----|Math programming\n");
        S = 0;
        for (i = 1; i <= n; i++) {
            mult = 1;
            for (j = 1; j <= i; j++) {
                mult *= ((j + 1) * sin(j));
            }
            S += (mult / (i * (i + 1)));
        }
        printf("Sum = [%9.7Lf]\n", S);
        printf("-----|Dynamic programming\n");
        S = 0; mult = 1;
        for (i = 1; i <= n; i++) {
            mult *= ((i + 1) * sin(i));
            S += (mult / (i * (i + 1)));
        }
        printf("Sum = [%9.7Lf]\n\n", S);
    }
    else printf("Error, during devision by 0(no decision has been made)\n");
    return 0;
}
```

Тестування

Вхідні дані n	Отримані значення з калькулятором	Отримані значення з першим алгоритмом	Отримані значення з другим алгоритмом
-1	Error	Error	Error
0	Error	Error	Error
1	0.841471	0.841471	0.841471
2	1.6066184	1.6066184	1.6066184
3	1.8225736	1.8225736	1.8225736
4	1.3322673	1.3322673	1.3322673
5	3.2129339	3.2129339	3.2129339

Розрахунок обчислення кількості операцій

Операція	К - позначення	Вкладені цикли	Динамічне програмування
Виконання заголовку зовнішнього циклу	K_{iL}	n	n
Виконання заголовку внутрішнього циклу	K_{jL}	$(n^2+n)/2$	0
Присвоєння	K_{as}	$((n^2+5n)/2)+1$	$2n+2$
Арифметичні операції	K_a	$(3n^2+11n)/2$	$7n$
Порівняння	K_c	1	1
Виклики стандартних функцій	K_F	$((n^2+n)/2)+2$	$n+2$
Загальна кількість	K	$3n^2+10n+4$	$11n+5$

Скріншоти вхідних даних та результатів обчислення

```
-----|Input your n: 1|
-----|Math programming
Sum = [0.8414710]
-----|Dynamic programming
Sum = [0.8414710]
```

```
1 #include <stdio.h>
2 #include <math.h>
3
4 int main() {
5     int i, j, n;
6     printf("-----|Input your n: ");
7     scanf("%d", &n);
8     if (n > 0) {
9         printf("-----|Math programming");
10        S = 0;
11        for (i = 1; i <= n; i++) {
12            S = S + 1/i;
```

```
-----|Input your n: 2|
-----|Math programming
Sum = [1.6066184]
-----|Dynamic programming
Sum = [1.6066184]
```

```
1 #include <stdio.h>
2 #include <math.h>
3
4 int main() {
5     int i, j, n;
6     printf("-----|Input your n: ");
7     scanf("%d", &n);
8     if (n > 0) {
9         printf("-----|Math programming");
10        S = 0;
11        for (i = 1; i <= n; i++) {
12            S = S + 1/i;
```

```
-----|Input your n: 3|
-----|Math programming
Sum = [1.8225736]
-----|Dynamic programming
Sum = [1.8225736]
```

```
1 #include <stdio.h>
2 #include <math.h>
3
4 int main() {
5     int i, j, n;
6     printf("-----|Input your n: ");
7     scanf("%d", &n);
8     if (n > 0) {
9         printf("-----|Math programming");
10        S = 0;
11        for (i = 1; i <= n; i++) {
12            S = S + 1/i;
```

```
-----|Input your n: 4|
-----|Math programming
Sum = [1.3322673]
-----|Dynamic programming
Sum = [1.3322673]
```

```
1 #include <stdio.h>
2 #include <math.h>
3
4 int main() {
5     int i, j, n;
6     printf("-----|Input your n: ");
7     scanf("%d", &n);
8     if (n > 0) {
9         printf("-----|Math programming");
10        S = 0;
11        for (i = 1; i <= n; i++) {
12            S = S + 1/i;
```

```
-----|Input your n: 5|
-----|Math programming
Sum = [3.2129339]
-----|Dynamic programming
Sum = [3.2129339]
```

```
1 #include <stdio.h>
2 #include <math.h>
3
4 int main() {
5     int i, j, n;
6     printf("-----|Input your n: ");
7     scanf("%d", &n);
8     if (n > 0) {
9         printf("-----|Math programming");
10        S = 0;
11        for (i = 1; i <= n; i++) {
12            S = S + 1/i;
```

```
-----|Input your n: -1|
Error, during devision by 0(no decision has been made)
```

```
-----|Input your n: 0|
Error, during devision by 0(no decision has been made)
```