

Студент: F№

Компоненти на оценяването: писмена част (програмен код, написан на този лист през първите 30 минути); практическа част (компилиране и изпълнение на код). Писмената част включва задача 1. Практическата част е върху останалите задачи. Изпълнението на практическата част се изпраща чрез Moodle като текстов файл, съдържащ изработения код. Оценяване: 0-30т.-2, 31-50т.-3, 51-70т.-4, 71-90т.-5, 91-100т.-6.

Задача 1. (25т.) Програма при която потребителя въвежда различни стойности за n и x , където $2 \leq n < 100$ и $-1 < x < 1$ за:

$$\begin{aligned} & (-1)^{(1+1)} \frac{1}{1} x^1 + (-1)^{(2+1)} \left(\frac{1}{1} + \frac{1}{2} \right) x^2 + (-1)^{(3+1)} \left(\frac{1}{1} + \frac{1}{2} + \frac{1}{3} \right) x^3 + \dots + (-1)^{(n+1)} \left(\frac{1}{1} + \frac{1}{2} + \frac{1}{3} + \frac{1}{4} + \dots + \frac{1}{n} \right) x^n \\ & = \sum_{m=1}^n (-1)^{(m+1)} \sum_{k=1}^m \left(\frac{1}{k} \right) x^m \end{aligned}$$

Да се пресметне и отпечата стойността на сумата по зададена от потребителя стойност на x от тип double между -1 и 1 (например $x = 0.5$) и да се определи и отпечата стойността на n за която сумата се приближава до стойността на израза $\log(1+x) / (1+x)$, като разликата да не е по-голяма от 0.0001 и за \log да се използва функцията от `<cmath>` на C++.

Задача 2. (40т.) Да се напише програма за:

1. За едномерен масив от цели числа с N на брой елемента. По ваш избор за конкретна стойност на N в интервала $12 \leq N \leq 100$. Да се използва `#define` за N и във всички цикли за работа с масива да се ползва N . Да се попълни масивът със случайни стойности в интервал от 10 до 99. Общият брой нечетни числа в масива да не е повече от $1/3$ от N . Да се отпечатаат елементите на масива с индекси и стойности.
2. За всеки елемент на масива да се отпечата индекс, стойност и броя на повторенията му в масива.
3. Да се направи избор на 8 случайни елемента от масива, които да се отпечатаат. За четен N брой елемента да се определи колко от тези 8 случайно избрани елемента са в първата половина на масива тоест индекси от 0 до $N/2$ и съответно във втората половина на масива с индекси от $N/2$ до $N-1$.

Задача 3. (35т.) Да се състави програма, в която потребителят въвежда 4 цифрено число за което: 1) проверка дали числото е просто 2) да се намери колко пъти всяка от цифрите [2...9] дели без остатък въведеното число.

// Задача 1

```
#include <iostream>
#include <iomanip>
#include <cmath>
```

```
using namespace std;
```

```
int main() {
```

```
int n = 2, sign = 1;
```

```
double total_sum = 0.0;
```

```
double temp_sum = 1.0;
```

```
double diff = 1.0, x = 0.5;
```

```
do {
```

```
cout << "Enter x(-1,1): ";
```

```
cin >> x;
```

```
//
```

```
} while (fabs(x) >= 1);
```

```
total_sum += x;
```

```
while (n <= 100 && diff >= 0.0001) {
```

```
temp_sum += 1.0 / n;
```

```
sign *= -1;
```

```
total_sum += sign * temp_sum * pow(x, n);
```

```
n++;
```

```
diff = fabs(total_sum - log(x+1) / (x+1));
```

```
cout << "n:" << setw(3) << n << " sum: " << setw(12) << total_sum
```

```
<< " log(" << x << ")/(" << x << "+1): " << log(x+1) / (x+1)
```

```
<< " difference: " << setw(12) << diff << endl;
```

```
}
```

```
return 0;
```

```
}
```

// Задача 2

```
#include <iostream>
#include <iomanip>
#include <cstdlib>
#include <ctime>
```

```
#define N 22
```

```
using namespace std;
```

```
int main() {
```

```
// Задача 2.1
```

```
int A[N], C[N] = {0}, R_index[8]={0};
```

```
int r, cnt = 0;
```

```
srand(time(0));
```

```
for(int i = 0; i < N; i++) {
```

```
    r = rand()%89 + 10 ;
```

```
    if (r%2 != 0) {
```

```
        cnt++;
```

```
    }
```

```
    if (cnt < N/3) {
```

```
        A[i] = r;
```

```
    }
```

```
    else {
```

```
        if (r%2!=0) {
```

```
            while(r%2!=0) {
```

```
                r = rand()%89 + 10;
```

```
            }
```

```
            A[i] = r;
```

```
        } else { A[i] = r; }
```

```
    }
```

```
}
```

```
for (int i=0;i<N;i++){
```

```
    for (int j=0;j<N;j++) {
```

```
        if (A[j] == A[i]) { C[i] += 1;}
```

```
    }
```

```
}
```

// Задача 2.2

```
cnt = 0;
for (int i = 0; i < N; i++) {
    cout << "A[" << setw(3) << i << " ] = " << A[i] << " Repeated: " << C[i] <<
endl;
    if (A[i]%2==0) {cnt++;}
}
cout << endl;
cout << "Total even numbers: " << cnt << endl;
cout << "Total odd numbers: " << N - cnt << endl;
cout << "\nSelect 8 random elements (unique indexes from array A):\n";
```

// Задача 2.3

```
int flag = 0;
cnt = 0;
// 8 случайно избрани елемента от масива A. За тази цел използваме
// помощен масив R_index в който записваме 8 случайни индекси без
// да се повторят за масива A
```

```
while (cnt < 8) {
    r = rand()%N;
    for (int i=0; i < cnt; i++) {
        if (R_index[i] == r) {
            flag = 1;
            break;
        }
        flag = 0;
    }
    if (flag == 0) { R_index[cnt] = r; cnt++; }
}
// отпечатване на случайно избраните индекси от масив A
```

```
cnt = 0;
for (int i=0; i<8; i++) {
    if (R_index[i] <= N / 2) { cnt++; }
    cout << "R_index[" << i << "] = " << setw(3) << R_index[i] << " "
        << setw(4) << "A[" << setw(2) << R_index[i] << " ] = "
        << A[R_index[i]] << endl;
}
cout << "\nRandom " << cnt << " indexes are from A[0] to " << "A["
    << N/2 << "];";
cout << "\nRandom " << 8-cnt << " indexes are from A[" << N/2 << "]" to "
    << "A[" << N-1 << "]\n";
```

```
return 0;
}
```

// Задача 3

```
#include <iostream>
```

```
using namespace std;
```

```
int main()
```

```
{ int num, digits[8] = {2,3,4,5,6,7,8,9};
```

// задача 3.1

```
do {
```

```
    cout << "Enter 4 digit integer number: ";
```

```
    if (!(cin >> num)) {
```

```
        cout << "You did not enter a correct integer number!" << endl;
```

```
        break;
```

```
    }
```

```
} while( num < 999 || num > 9999);
```

```
for (int i = 2; i <= num; i++) {
```

```
    if (num % i == 0) {
```

```
        if (i == num) {
```

```
            cout << num << " is a PRIME NUMBER \n";
```

```
        } else {
```

```
            cout << num << " is NOT A PRIME NUMBER \n";
```

```
            break;
```

```
        }
```

```
    }
```

```
}
```

// задача 3.2

```
for (int i=0; i<8; i++) {
```

```
    int k = 0, r = num;
```

```
    do {
```

```
        if (r % digits[i] == 0) {
```

```
            r = r / digits[i];
```

```
            k++;
```

```
        } else { break; }
```

```
    } while (1);
```

```
    cout << "Digit " << digits[i] << " divides " << k << " times number " << num << endl;
```

```
}
```

```
return 0;
```

```
}
```