|  |
| --- |
| МИНОБРНАУКИ РОССИИ |
| Федеральное государственное бюджетное образовательное учреждение  высшего образования  **«МИРЭА – Российский технологический университет»**  **РТУ МИРЭА** |
| Институт искусственного интеллекта |
| Кафедра Технологий Искусственного Интеллекта |

Практическая работа № 9

по дисциплине

«Процедурное программирование»

Обучающийся: Погосян С. А.

Группа: КВБО-07-23

Руководитель *Яковлев Д. А*

Москва 2023

**Описание работы:**

Выделите память для массива чисел и заполнить его любым способом. Создайте функцию, выполняющую задачу, указанную в варианте. Вызовите её из основной функции, получите её результат и выведите его.

**Задание:**

****

**#include <stdio.h>**

**#include <time.h>**

**#include <stdlib.h>**

**#include <math.h>**

**#define N 20**

**int get\_3k(int \*ar) {**

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

**if ((abs(ar[i]) - 1) % 3 == 0) {**

**printf("%3d ", ar[i]);**

**}**

**}**

**return 0;**

**}**

**int main() {**

**srand(time(NULL));**

**int \*ar;**

**ar = (int \*)malloc(N \* sizeof(int));**

**printf("Initial Array:\n ");**

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

**ar[i] = rand() % (100 + 100 + 1) - 100;**

**printf("%3d ", ar[i]);**

**}**

**putchar('\n');**

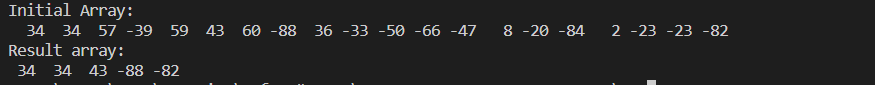
**printf("Result array:\n");**

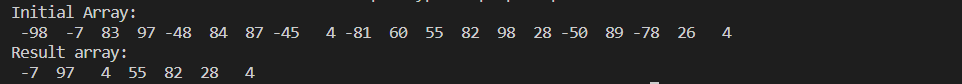
**get\_3k(ar);**

**free(ar);**

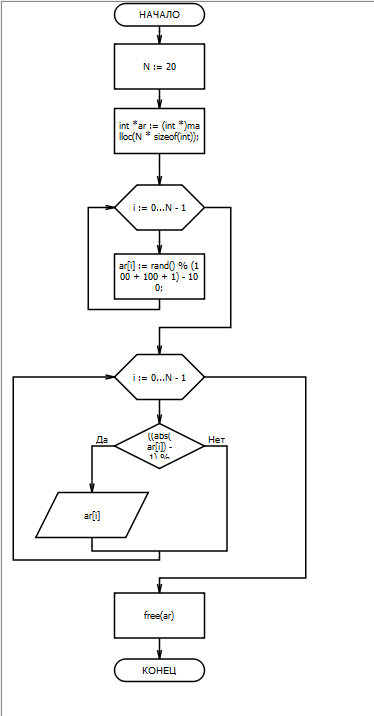
**}**

**Результат работы программы:**





**Алгоритм в виде блок схемы:**

****

**Задание:**



**Код программы:**

**#include <stdio.h>**

**#include <time.h>**

**#include <stdlib.h>**

**#include <math.h>**

**#define N 10**

**int get\_total(int \*ar) {**

**int total = 0;**

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

**if (ar[i] % 3 == 0) {**

**total += i;**

**}**

**}**

**return total;**

**}**

**int main() {**

**srand(time(NULL));**

**int \*ar;**

**ar = (int \*)malloc(sizeof(int) \* N);**

**int total = 0;**

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

**ar[i] = rand() % 10;**

**printf("%3d ", ar[i]);**

**}**

**total = get\_total(ar);**

**printf("\ntotal = %d\n", total);**

**return 0;**

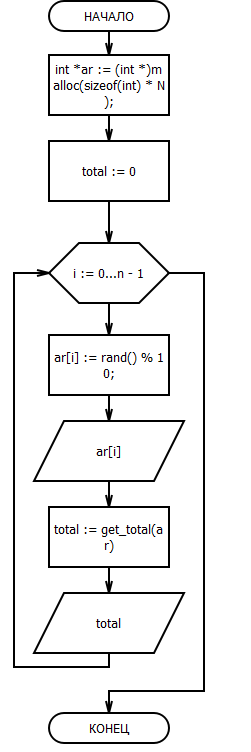
**}**

**Результат работы программы:**

****



**Алгоритм в виде блок-схемы:**



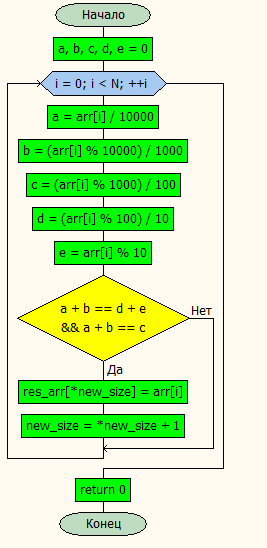
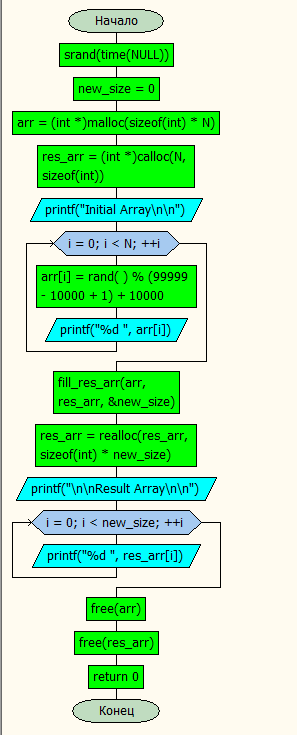
**Задание:**



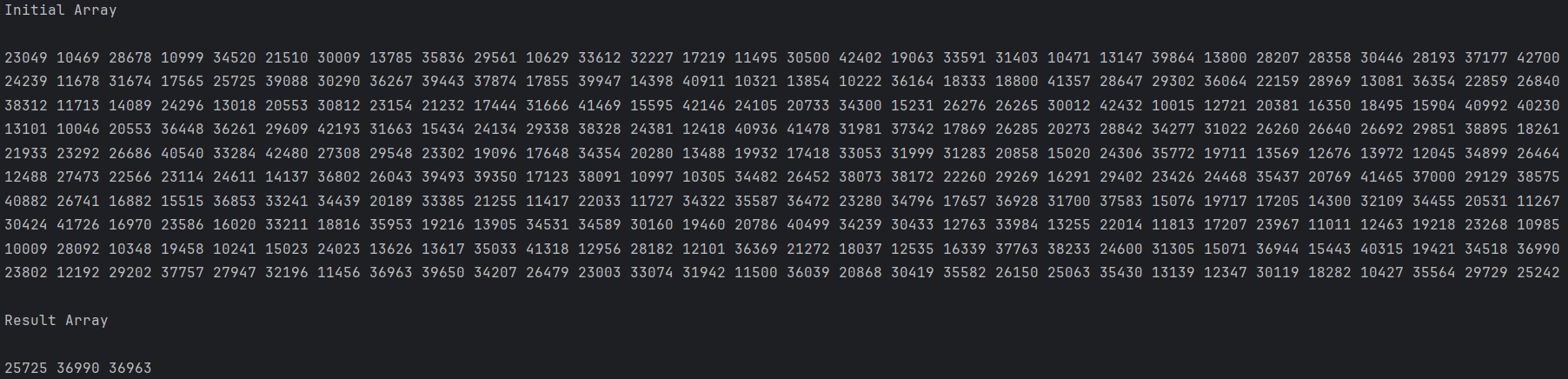
**Код программы:**

#include <stdio.h>  
#include <time.h>  
#include <stdlib.h>  
  
#define N 300  
  
int fill\_res\_arr(int \*arr, int \*res\_arr, int \*new\_size) {  
 int a, b, c, d, e = 0;  
 for (int i = 0; i < N; ++i) {  
 a = arr[i] / 10000;  
 b = (arr[i] % 10000) / 1000;  
 c = (arr[i] % 1000) / 100;  
 d = (arr[i] % 100) / 10;  
 e = arr[i] % 10;  
 if (a + b == d + e && a + b == c) {  
 res\_arr[\*new\_size] = arr[i];  
 \*new\_size = \*new\_size + 1;  
 }  
 }  
  
 return 0;  
}  
  
int main() {  
  
 srand(time(NULL));  
  
  
 int new\_size = 0;  
  
 int \*arr = (int \*) malloc(sizeof(int) \* N);  
 int \*res\_arr = (int \*) calloc(N, sizeof(int));  
  
 printf("Initial Array\n\n");  
  
 for (int i = 0; i < N; ++i) {  
 arr[i] = rand() % (99999 - 10000 + 1) + 10000;  
 printf("%d ", arr[i]);  
  
 }  
 fill\_res\_arr(arr, res\_arr, &new\_size);  
  
 res\_arr = realloc(res\_arr, sizeof(int) \* new\_size);  
  
 printf("\n\nResult Array\n\n");  
  
 for (int i = 0; i < new\_size; ++i) {  
 printf("%d ", res\_arr[i]);  
 }  
  
 free(arr);  
 free(res\_arr);  
  
  
 return 0;  
}

**Алгоритм в виде блок-схемы:**

** **

**Результат работы программы:**

****