

МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ РОССИЙСКОЙ ФЕДЕРАЦИИ
федеральное государственное автономное образовательное
учреждение высшего образования
НАЦИОНАЛЬНЫЙ ИССЛЕДОВАТЕЛЬСКИЙ
ТОМСКИЙ ПОЛИТЕХНИЧЕСКИЙ УНИВЕРСИТЕТ

Инженерная школа информационных технологий и робототехники
Отделение автоматизации и робототехники
Направление мехатроника и робототехника

Отчет
по лабораторной работе №5
по дисциплине
«Основы программирования и алгоритмизации»
Структуры в C++

Выполнил:

Студент группы 8E21

Н.С. Моисеев

Проверил:

Ассистент ОАР ИШИТР

Я.О. Кургинов

Томск 2023

Вариант 10

Цель работы:

Поработать со структурами в C++.

Задание 1

При заказе в KFC формируется электронная очередь. Если составляющие заказа готовы, то заказ собирается и отдается покупателю. Если нет, то пропускается и собирается следующий.

Реализовать структуру данных, представляющую электронную очередь.

Алгоритмы должны выполнять добавление в очередь и удаление из очереди при получении заказа, выбор заказа, который собирается на данный момент, исходя из порядка в очереди и готовности позиций заказа. Учесть что кухня периодически пополняет запас готовых позиций.

Блок схема (Рисунок 1)

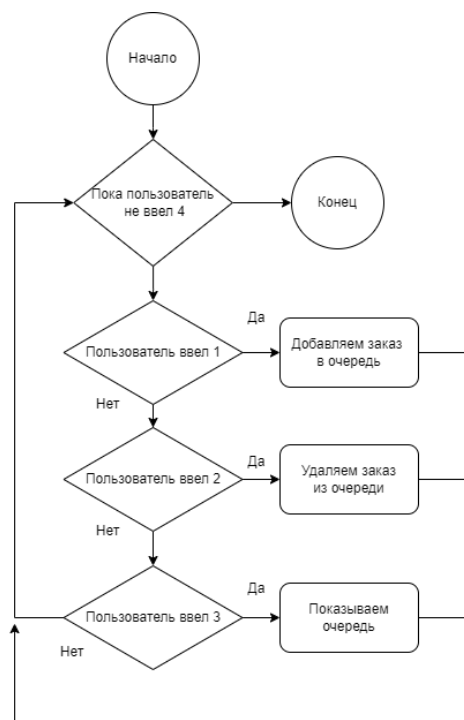


Рисунок 1 - Блок-схема к заданию 1.

Текст программы (Листинг 1, Приложение А)

Результаты работы (Рисунок 2)

```
1) Insert human to queue tp choose what to do
2) Delete human from queue
3) Display all the elements of queue
4) Exit
Enter your choice :
1
Insert count of hamburgers and Cola Dobriy :
2 3
Enter your choice :
1
Insert count of hamburgers and Cola Dobriy :
4 2
Enter your choice :
3
Queue elements are: 2 Haburgers 3 Cola Dobriy ; 4 Haburgers 2 Cola Dobriy ;
Enter your choice :
2
Element deleted from queue is : 2 3
Enter your choice :
3
Queue elements are: 4 Haburgers 2 Cola Dobriy ;
Enter your choice :
4
Exit
```

Рисунок 2 - Скриншот результата работы программы к заданию 1 на консоли.

Вывод:

В результате лабораторной работы были решены поставленные задачи со структурами, используя C++.

Приложение А

Листинг 1 – Программа к заданию 1.

```
1. #include <iostream>
2.
3. using namespace std;
4.
5. struct node {
6.     int data;
7.     int data1;
8.     struct node *next;
9. };
10. struct node* front = NULL;
11. struct node* rear = NULL;
12. struct node* temp;
13.
14. struct queue{
15.     int kfccombo;
16.     int cola;
17. };
18.
19. void Insert() {
20.     int val;
21.     int val1;
22.     cout<<"Insert count of hamburgers and Cola Dobriy : "<<endl;
23.     cin>>val;
```

```

24.  cin>>val1;

25.  if (rear == NULL) {

26.      rear = (struct node *)malloc(sizeof(struct node));

27.      rear->next = NULL;

28.      rear->data = val;

29.      rear->data1 = val1;

30.      front = rear;

31.  } else {

32.      temp=(struct node *)malloc(sizeof(struct node));

33.      rear->next = temp;

34.      temp->data = val;

35.      temp->data1 = val1;

36.      temp->next = NULL;

37.      rear = temp;

38.  }

39. }

40. void Delete() {

41.     temp = front;

42.     if (front == NULL) {

43.         cout<<"Underflow"<<endl;

44.         return;

45.     }

46.     else

47.     if (temp->next != NULL) {

48.         temp = temp->next;

49.         cout<<"Element deleted from queue is : "<<front->data << " " << front-
>data1<<endl;

50.         free(front);

51.         front = temp;

52.     } else {

```

```

53.     cout<<"Element deleted from queue is : "<<front->data << " " << front->data1
        <<endl;

54.     free(front);

55.     front = NULL;

56.     rear = NULL;

57. }

58. }

59. void Display() {

60.     temp = front;

61.     if ((front == NULL) && (rear == NULL)) {

62.         cout<<"Queue is empty"<<endl;

63.         return;

64.     }

65.     cout<<"Queue elements are: ";

66.     while (temp != NULL) {

67.         int x = temp->data;

68.         if(x!=0) cout << temp->data << " Haburgers" << " ";

69.         if(x!=0) cout << temp->data1 << " Cola Dobriy" << " ";

70.         cout << "; ";

71.         temp = temp->next;

72.     }

73.     cout<<endl;

74. }

75. int main() {

76.     int ch;

77.     cout<<"1) Insert human to queue tp choose what to do"<<endl;

78.     cout<<"2) Delete human from queue"<<endl;

79.     cout<<"3) Display all the elements of queue"<<endl;

80.     cout<<"4) Exit"<<endl;

81.     do {

```

```
82.     cout<<"Enter your choice : "<<endl;
83.     cin>>ch;
84.     switch (ch) {
85.         case 1: Insert();
86.         break;
87.         case 2: Delete();
88.         break;
89.         case 3: Display();
90.         break;
91.         case 4: cout<<"Exit"<<endl;
92.         break;
93.         default: cout<<"Invalid choice"<<endl;
94.     }
95. } while(ch!=4);
96. return 0;
97. }
```