

```
#include <iostream>
using namespace std;
```

```
int subtract() {
    int a, b;
    cout << "Enter two numbers to subtract: ";
    cin >> a >> b;
    cout << "The difference is: " << a - b << endl;
    return 0;
}
```

```
int Multiply() {
    int a, b;
    cout << "Enter two numbers to multiply: ";
    cin >> a >> b;
    cout << "The product is: " << a * b << endl;
    return 0;
}
```

```
int Divide() {
    int a, b;
    cout << "Enter two numbers to divide: ";
    cin >> a >> b;
    if (b != 0) {
        cout << "The quotient is: " << a / b << endl;
    } else {
        cout << "Error: Division by zero is not allowed." << endl;
    }
    return 0;
}
```

```
int sum() {
    int a, b;
    cout << "Enter two numbers to add: ";
    cin >> a >> b;
    cout << "The sum is: " << a + b << endl;
    return 0;
}
```

```
int menu() {  
    int n;  
    cout << "Menu Options:" << endl;  
    cout << "1. Add 2 Numbers" << endl;  
    cout << "2. Subtract 2 Numbers" << endl;  
    cout << "3. Multiply 2 Numbers" << endl;  
    cout << "4. Divide 2 Numbers" << endl;  
    cout << "5. Exit" << endl;  
    cout << "Please select an option (1-5): ";  
    cin >> n;  
    return n;  
}
```

```
void menuOption() {  
    while (true) {  
        int option = menu();  
  
        if (option == 1) {  
            sum();  
        }  
        else if (option == 2) {  
            subtract();  
        }  
        else if (option == 3) {  
            Multiply();  
        }  
        else if (option == 4) {  
            Divide();  
        }  
        else if (option == 5) {  
            cout << "Exiting the program." << endl;  
            break;  
        }  
        else {  
            cout << "Error message. Please select a valid option (1-5)." << endl;  
        }  
    }  
}
```

```
3 void menuOption() {
4     while (true) {
5         int option = menu();
6
7         if (option == 1) {
8             sum();
9         }
10        else if (option == 2) {
11            subtract();
12        }
13        else if (option == 3) {
14            Multiply();
15        }
16        else if (option == 4) {
17            Divide();
18        }
19        else if (option == 5) {
20            cout << "Exiting the program." << endl;
21            break;
22        }
23        else {
24            cout << "Error message. Please select a valid option (1-5)." << endl;
25        }
26    }
27 }
28
29 int main() {
30     menuOption();
31     return 0;
32 }
```

```
#include <iostream>
using namespace std;
int max3(int a, int b, int c){
    int m = a;
    if (b > m) {
        m = b;
    }
    if (c > m) {
        m = c;
    }
    return m;
}
int max4(int a, int b, int c, int d){
    int m = max3(a,b,c);
    if (m > d) {
        return m;
    } else {
        return d;
    }
}
int max5(int a, int b, int c, int d, int e){
    int m = max4(a,b,c,d);
    if (m > e) {
        return m;
    } else {
        return e;
    }
}
int max6(int a, int b, int c, int d, int e, int f){
    int m = max5(a,b,c,d,e);
    if (m > f) {
        return m;
    } else {
        return f;
    }
}
int main() {
    int a,b,c,d,e,f;
    cout << "Enter six numbers: ";
    cin >> a >> b >> c >> d >> e >> f;
    int res = max6(a,b,c,d,e,f);
    cout << "The maximum number is: " << res << endl;
    return 0;
}
```

```
1  #include <iostream>
2  #include <string>
3  using namespace std;
4  string rev_string(const string &str ) {
5      string res = "";
6      for (int i = str.length() - 1; i >= 0 ; i--) {
7          res += str[i];
8      }
9      return res;
10 }
11
12 int main() {
13     string input;
14     cout << "Enter a string: ";
15     cin >> input;
16     string reversed = rev_string(input);
17     cout << "Reversed string: " << reversed << endl;
18     return 0;
19 }
20
```

```
1  #include <iostream>
2  #include <string>
3  #include <algorithm>
4  using namespace std;
5  bool ispalindrome(int arr[], int n) {
6      for (int i = 0; i < n / 2; i++) {
7          if (arr[i] != arr[n - i - 1]) {
8              return false;
9          }
10     }
11     return true;
12 }
13 int main() {
14     int arr[1000];
15     int n;
16     cout << "Enter the number of elements in the array: ";
17     cin >> n;
18     cout << "Enter the elements of the array: ";
19     for (int i = 0; i < n; i++) {
20         cin >> arr[i];
21     }
22     if (ispalindrome(arr, n)) {
23         cout << "The array is a palindrome." << endl;
24     } else {
25         cout << "The array is not a palindrome." << endl;
26     }
27     return 0;
28 }
29
```

```
1  #include <iostream>
2  #include <string>
3  #include <algorithm>
4  #include <cmath>
5  using namespace std;
6  void set_powers(int arr[], int len , int m ) {
7      int n = 0;
8      for (int i = 0; i <= len; i++) {
9          arr[i] = pow(m, i);
10     }
11 }
12 int main() {
13     int len, m;
14     cout << "Enter the Number: ";
15     cin >> len;
16     cout << "Enter the Power: ";
17     cin >> m;
18     int arr[len];
19     set_powers(arr, len, m);
20     cout << "The array Is: ";
21     for (int i = 0; i < len; i++) {
22         cout << arr[i] << " ";
23     }
24     return 0;
25 }
26
```

```
1  #include <iostream>
2  #include <string>
3  #include <algorithm>
4  #include <cmath>
5  using namespace std;
6  bool is_prime(int num) {
7      if (num <= 1) return false;
8      for (int i = 2; i * i <= num; i++) {
9          if (num % i == 0)
10             return false;
11     }
12     return true;
13 }
14 int nth_prime(int n) {
15     int count = 0;
16     int num = 2;
17     while (true) {
18         if (is_prime(num)) {
19             count++;
20             if (count == n) {
21                 return num;
22             }
23         }
24         num++;
25     }
26 }
27
28 int main() {
29     int n;
30     cout << "Enter the position of the prime number: ";
31     cin >> n;
32     int prime = nth_prime(n);
33     cout << "The " << n << "-th prime number is: " << prime << endl;
34
35     return 0;
36 }
```



```

1  #include <iostream>
2  #include <string>
3  #include <algorithm>
4  #include <cmath>
5  using namespace std;
6  bool starts_with(string input, string pattern, int pos) {
7      for (int i = 0; i < pattern.length(); i++) {
8          if (pos + i >= input.length() || input[pos + i] != pattern[i]) {
9              return false;
10         }
11     }
12     return true;
13 }
14 string replace_str(string input, string pattern, string to) {
15     string result = "";
16     int i = 0;
17     while (i < input.length()) {
18         if (starts_with(input, pattern, i)) {
19             result += to;
20             i += pattern.length();
21         } else {
22             result += input[i];
23             i++;
24         }
25     }
26     return result;
27 }
28 int main() {
29     string input, pattern, to;
30     cout << "Enter the input string: ";
31     cin >> input;
32     cout << "Enter the pattern to replace: ";
33     cin >> pattern;
34     cout << "Enter the replacement string: ";
35     cin >> to;
36     string output = replace_str(input, pattern, to);
37     cout << "Result: " << output << endl;
38     return 0;
39 }

```

```

1  #include <iostream>
2  #include <string>
3  using namespace std;
4
5  const int max_specialization = 20;
6  const int max_queue = 5;
7
8  struct Patient {
9      string name;
10     int status; };
11
12 Patient hospital[max_specialization + 1][max_queue];
13 int count[max_specialization + 1] = {0};
14
15 void Add() {
16     int spec, status;
17     string name;
18
19     cout << "Enter specialization (1-20): ";
20     cin >> spec;
21
22     if (spec < 1 || spec > 20) {
23         cout << "Invalid specialization.\n";
24         return;
25     }
26
27     if (count[spec] >= max_queue) {
28         cout << "Sorry, no available slots in this specialization.\n";
29         return;
30     }
31
32     cout << "Enter patient name: ";
33     cin >> name;
34     cout << "Enter status (0 = regular, 1 = urgent): ";
35     cin >> status;
36
37     if(status == 1) {
38         for (int i = count[spec]; i < max_queue; i++) {

```

```

37     if (status == 1) {
38         for (int i = count[spec]; i > 0; i--) {
39             hospital[spec][i] = hospital[spec][i - 1];
40         }
41         hospital[spec][0] = {name, status};
42     } else {
43         hospital[spec][count[spec]] = {name, status};
44     }
45
46     count[spec]++;
47     cout << "Patient added successfully.\n";
48 }
49 void Print() {
50     for (int spec = 1; spec <= max_specialization; spec++) {
51         if (count[spec] > 0) {
52             cout << "Specialization " << spec << ":\n";
53             for (int i = 0; i < count[spec]; i++) {
54                 cout << "    [Spec " << spec << "] " << hospital[spec][i].name;
55                 if (hospital[spec][i].status == 1)
56                     cout << " (urgent)";
57                 cout << "\n";
58             }
59         }
60     }
61 }
62
63
64 void Get() {
65     int spec;
66     cout << "Enter specialization (1-20): ";
67     cin >> spec;
68
69     if (spec < 1 || spec > 20 || count[spec] == 0) {
70         cout << "No patients in this specialization.\n";
71         return;
72     }

```

```

64 void Get() {
65     int spec;
66     cout << "Enter specialization (1-20): ";
67     cin >> spec;
68
69     if (spec < 1 || spec > 20 || count[spec] == 0) {
70         cout << "No patients in this specialization.\n";
71         return;
72     }
73
74     cout << hospital[spec][0].name << ", please go with the doctor.\n";
75
76     for (int i = 1; i < count[spec]; i++) {
77         hospital[spec][i - 1] = hospital[spec][i];
78     }
79
80     count[spec]--;
81 }
82 void Choice() {
83     int choice;
84     while (true) {
85         cout << "\nEnter your choice:\n";
86         cout << "1. Add new patient\n";
87         cout << "2. Print all patients\n";
88         cout << "3. Get new patient\n";
89         cout << "4. Exit\n";
90         cout << "Your choice: ";
91         cin >> choice;
92         if (choice == 1){
93             Add();
94         } else if (choice == 2){
95             Print();
96         } else if (choice == 3){
97             Get();
98         } else if (choice == 4){
99             cout << "Exiting the program.\n";
100            break;

```

```
count[spec]--;
```

```
}
```

```
void Choice() {
```

```
    int choice;
```

```
    while (true) {
```

```
        cout << "\nEnter your choice:\n";
```

```
        cout << "1. Add new patient\n";
```

```
        cout << "2. Print all patients\n";
```

```
        cout << "3. Get new patient\n";
```

```
        cout << "4. Exit\n";
```

```
        cout << "Your choice: ";
```

```
        cin >> choice;
```

```
        if (choice == 1){
```

```
            Add();
```

```
        } else if (choice == 2){
```

```
            Print();
```

```
        } else if (choice == 3){
```

```
            Get();
```

```
        } else if (choice == 4){
```

```
            cout << "Exiting the program.\n";
```

```
            break;
```

```
        } else {
```

```
            cout << "Invalid choice. Please try again.\n";
```

```
        }
```

```
    }
```

```
}
```

```
int main() {
```

```
    Choice();
```

```
    return 0;
```

```
}
```