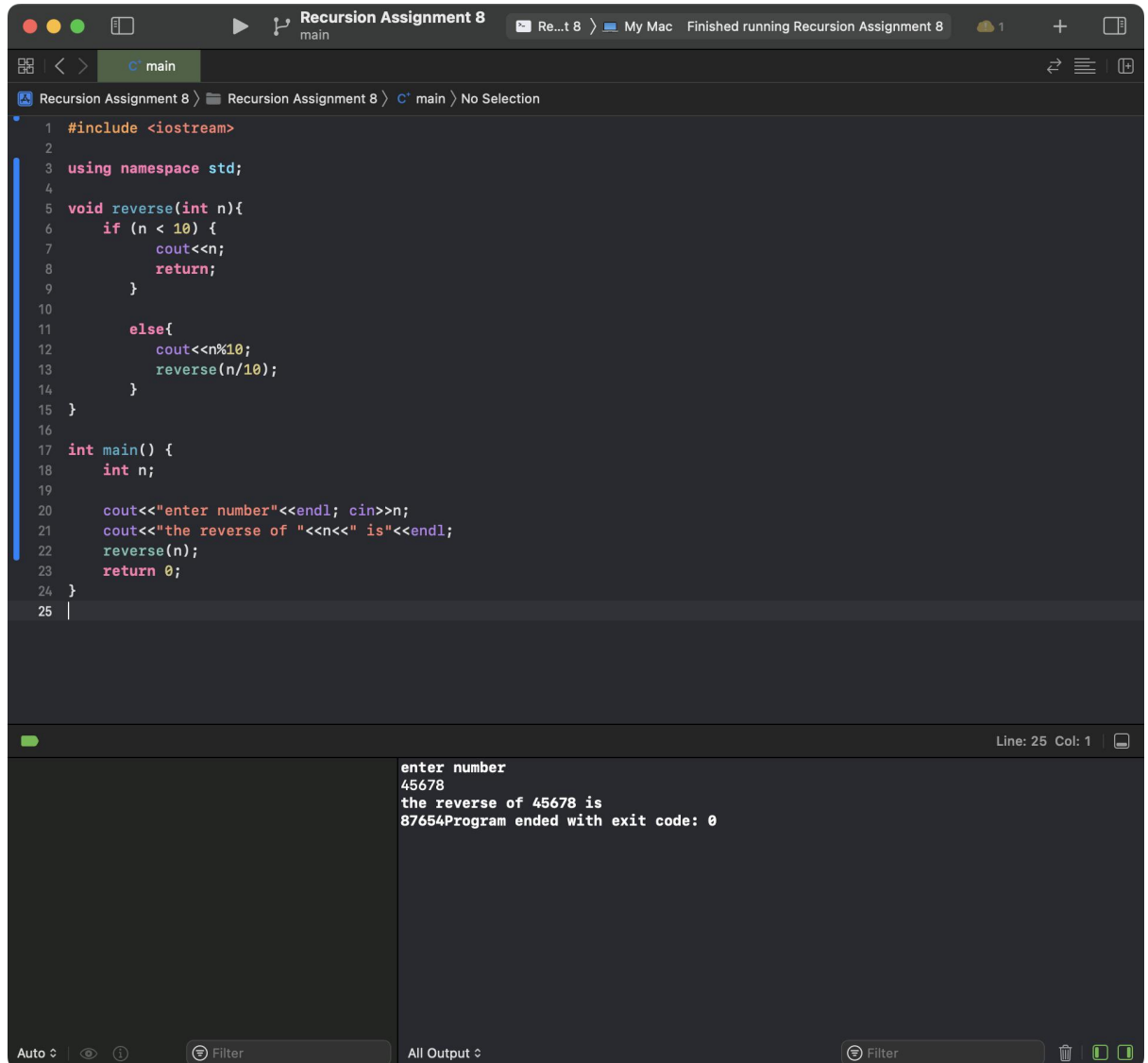


## HOMEWORK 8

### RECURSIVE FUNCTION

1. Create a program to reverse a number recursively.

For example, the number 7895 is reversed to 5987.



The screenshot shows a C++ IDE window titled "Recursion Assignment 8". The code is as follows:

```
1 #include <iostream>
2
3 using namespace std;
4
5 void reverse(int n){
6     if (n < 10) {
7         cout<<n;
8         return;
9     }
10
11     else{
12         cout<<n%10;
13         reverse(n/10);
14     }
15 }
16
17 int main() {
18     int n;
19
20     cout<<"enter number"<<endl; cin>>n;
21     cout<<"the reverse of "<<n<<" is"<<endl;
22     reverse(n);
23     return 0;
24 }
25 |
```

The output window at the bottom shows the following text:

```
enter number
45678
the reverse of 45678 is
87654Program ended with exit code: 0
```

2. Binomial coefficient can be defined recursively as follows

$$\binom{n}{0} = 1$$

$$\binom{n}{n} = 1$$

$$\binom{n}{k} = \binom{n-1}{k-1} + \binom{n-1}{k}$$

Create a program to calculate the binomial coefficient using a recursive function where n and k are user defined.

```

1  #include <iostream>
2
3  using namespace std;
4
5  int binomial(int n, int k){
6      if (k > n)
7          return 0;
8      if (k == 0 || k == n)
9          return 1;
10
11     return binomial(n - 1, k - 1) + binomial(n - 1, k);
12 }
13
14 int main() {
15     int n, k;
16
17     cout<<"insert n and k for binomial coefficient "; cin>>n>>k;
18     cout<<"\nValue of ("<<n<< ", "<<k<<") is "<< binomial(n, k);
19     cout<<endl;
20     return 0;
21 }
22

```

Line: 19 Col: 16

```

insert n and k for binomial coefficient 5 2

Value of (5, 2) is 10
Program ended with exit code: 0

```

3. The Ackerman function is recursively defined as follows.

$a(m,n) = n+1$  if  $m=0$   
 $a(m,n) = a(m-1,1)$  if  $m \neq 0$  and  $n=0$   
 $a(m,n) = a(m-1, a(m, n-1))$  if  $m \neq 0$  and  $n \neq 0$

Create a program to calculate the Ackerman function.

```

1  #include <iostream>
2
3  using namespace std;
4
5  int ackerman(int m, int n){
6      if (m == 0){
7          return n+1;
8      } else if ((m > 0) && (n == 0)){
9          return ackerman(m-1, 1);
10     } else if ((m > 0) && (n > 0)){
11         return ackerman(m-1, ackerman(m, n-1));
12     } else {
13         return 1;
14     }
15 }
16
17 int main() {
18     int m, n;
19
20     cout<<"insert m and n for Ackerman Function "; cin>>m>>n;
21     cout<<"\nValue of a("<m<<"<n<<" is "<< ackerman(m, n);
22     cout<<endl;
23     return 0;
24 }
25

```

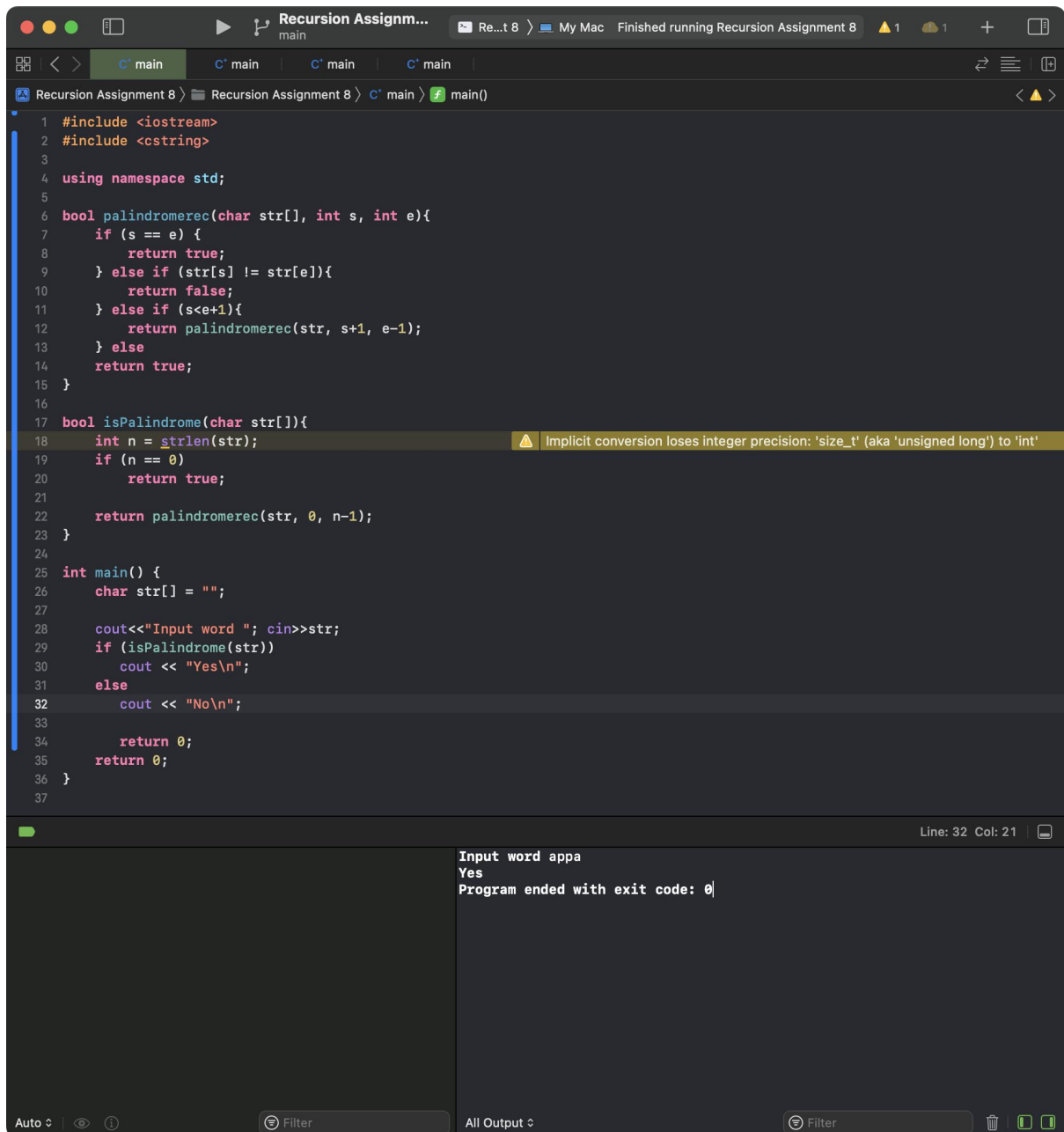
insert m and n for Ackerman Function 1 2

Value of a(1, 2) is 4

Program ended with exit code: 0

4. A word is *palindrome* if the word read the same backward as forward. For example, the words “a”, “refer”, and “noon” are palindrome, but the words “divided” and “winning” are not.

Create a program to determine if a word from user's input is palindrome using recursive function.



```
1 #include <iostream>
2 #include <cstring>
3
4 using namespace std;
5
6 bool palindromerec(char str[], int s, int e){
7     if (s == e) {
8         return true;
9     } else if (str[s] != str[e]){
10        return false;
11    } else if (s<e+1){
12        return palindromerec(str, s+1, e-1);
13    } else
14        return true;
15 }
16
17 bool isPalindrome(char str[]){
18     int n = strlen(str);
19     if (n == 0)
20         return true;
21
22     return palindromerec(str, 0, n-1);
23 }
24
25 int main() {
26     char str[] = "";
27
28     cout<<"Input word "; cin>>str;
29     if (isPalindrome(str))
30         cout << "Yes\n";
31     else
32         cout << "No\n";
33
34     return 0;
35 }
36
37
```

Line: 32 Col: 21

Input word appa  
Yes  
Program ended with exit code: 0