

# CSC 256 - Machine Structures

## Project 4

Total Points: 90 Points

## Description

For project four, your objective is to convert the given C++ code into MIPS assembly. Please do not modify the C++ code itself. You are only allowed to make modifications to the assembly file. Start writing your code below the `main: label` and above the `exit: label`. For this project stay BETWEEN these labels.

When doing a C++ to MIPS conversion, it can be done in the following steps:

- 1 Assign variables to registers. When inspecting code, any constant values in expressions may need to be assigned to temporary registers.
- 2 Initialize variables to registers. (actually put the values into the registers.)
- 3 Then move onto the rest of the code.

Before you begin, please make sure you click the link on ilearn to create your GitHub repo. After created please clone this repo with the `git clone repo_url` command.

## Expected Output:

```
Sum: 7179
Reversed Number: 98654
is Palindrome: 1
```

## Submission

When you have completed the assignment please commit all work done to your private repository. This can be done with the following commands:

```
git add .
git commit -m "some message"
git push
```

.

# Base MIPS Code

```
1 .data
2     endl:      .asciiz  "\n"    # used for cout << endl;
3     sumlbl:    .asciiz  "Sum: "  # label for sum
4     revlbl:    .asciiz  "Reversed Number: " # label for rev
5     pallbl:    .asciiz  "Is Palindrome: " # label for isPalindrome
6     arr:       .word 1
7               .word 2
8               .word 3
9               .word 4
10            .word 5
11            .word 4
12            .word 3
13            .word 2
14            .word 1
15     sumarr:    .word 1
16               .word 3
17               .word 44
18               .word 66
19               .word 88
20               .word 90
21               .word 9
22               .word 232
23               .word 4325
24               .word 2321
25 .text
26
27 # sum          —> $s0
28 # size         —> $s1
29 # address of sumarr —> $s2
30 # rev          —> $s3
31 # num          —> $s4
32 # isPalindrome —> $s5
33 # address of arr —> $s6
34 # i            —> $t0
35 # beg          —> $s7
36 # end          —> $s8
37 # d            —> $t2
38 # 10           —> $t3
39 # 100          —> $t4
40 main:
41
42 exit:
43     la    $a0, sumlbl    # puts sumlbl into arg0 (a0 register) for cout
44     addi  $v0, $0, 4     # puts 4 in v0 which denotes we are printing a string
45     syscall              # make a syscall to system
46
47     move  $a0, $s0       # puts sum into arg0 (a0 register) for cout
48     addi  $v0, $0, 1     # puts 1 in v0 to denote we are printing an int
49     syscall              # make a syscall to system
```

```

50
51 la    $a0, endl          # puts the address of the string endl into a0
52 addi $v0, $0, 4          # puts 4 into v0 saying we are printing a string
53 syscall
54
55 la    $a0, revlbl         # puts revlbl into arg0 (a0 register) for cout
56 addi $v0, $0, 4          # puts 4 in v0 which denotes we are printing an string
57 syscall                  # make a syscall to system
58
59 move $a0, $s1             # puts rev into arg0 (a0 register) for cout
60 addi $v0, $0, 1          # puts 1 in v0 to denote we are printing an int
61 syscall                  # make a syscall to system
62
63 la    $a0, endl          # puts the address of the string endl into a0
64 addi $v0, $0, 4          # puts 4 into v0 saying we are printing a string
65 syscall
66
67 la    $a0, pallbl         # puts pallbl into arg0 (a0 register) for cout
68 addi $v0, $0, 4          # puts 4 in v0 which denotes we are printing a string
69 syscall                  # make a syscall to system
70
71 move $a0, $s3             # puts isPalindrome into arg0 (a0 register) for cout
72 addi $v0, $0, 1          # puts 1 in v0 to denote we are printing an int
73 syscall                  # make a syscall to system
74
75 la    $a0, endl          # puts the address of the string endl into a0
76 addi $v0, $0, 4          # puts 4 into v0 saying we are printing a string
77 syscall
78
79
80 addi $v0, $0, 10
81 syscall

```

p4codeBase.s

# C++ Equivalent

```
1 #include <iostream>
2
3 using namespace std;
4
5
6
7 int main(void)
8 {
9
10     int sum = 0;
11     int size = 10;
12     int sumarr[] = {1,3,44,66,88,90,9,232,4325,2321};
13     for(int i = 0; i < size; i++){
14         sum = sum + sumarr[i];
15     }
16
17     int num = 45689;
18     int rev = 0;
19     int d = -1;
20     while( num > 0){
21         d = num % 10;
22         rev = rev*10 + d;
23         num = num / 10;
24     }
25
26     int arr[] = {1,2,3,4,5,4,3,2,1};
27     int beg = 0;
28     int end = 8;
29     int isPalindrome = 1;
30     while(beg < end){
31         if (arr[beg] != arr[end]){
32             isPalindrome = -1;
33             break;
34         }
35         beg++;
36         end--;
37     }
38
39
40
41     cout << "Sum: " << sum << endl;
42     cout << "Reversed Number: " << rev << endl;
43     cout << "Is Palindrome: " << isPalindrome << endl;
44     return 0;
45 }
```

p4code.cpp