PPL – Homework 4

Zhitao Gong

Due on Nov 3

Abstract

Homework 4 covers subprogram design and implementation issues.

1 Parameter-Passing Method

1. (**30 points**) Consider the following `C++` snippet.

```
void swap(int a, int b)
{
  int tmp = a;
  a = b;
  b = tmp;
}

int main()
{
  int val = 2;
  int arr[] = {1, 3, 5, 7, 9};

  swap(val, arr[0]);
  swap(arr[0], arr[1]);
  swap(val, arr[val]);
}
```

   What are the values of `val` and `arr` after each call to `swap` respectively if assuming (a) pass by value, (b) pass by reference, (c) pass by value-result.

2. (**30 points**) Consider the following `C++` snippet.

```
#include <cstdio>
```

```
int n;

void foo(int k)
{
  n += 1;
  k += 4;
  printf("%d\n", n);
}

int main()
{
  n = 0;
  foo(n);
  printf("%d\n", n);
}
```

What are the outputs respectively if assuming (a) pass by value, (b) pass by reference, (c) pass by value-result.

3. (**20 points**) Consider the following `Javascript` code snippet.

```
var n;

function foo(k) {
  console.log(k);                // print k
  n += 1;
  console.log(k);                // print k
}

n = 0;
foo(n + 10);
```

What are the outputs respectively if assuming (a) pass by value, (b) pass by name.

4. (**10 points**) What are at least two arguments against the use of pass-by-name parameters?

2 Nested Subprogram

1. (**10 points**) Speculate on the issue of allowing nested subprograms in programming languages – why are they not allowed in many contemporary languages?