CvP - Werkcollege 4

Exercise 1 Explain the difference between compile time and run time. Also explain what this has to do with the terms 'static' and 'dynamic'.

Exercise 2 Consider the following C++ code:

```
bool dolphinsLeaveYet = false;

class fish {
  public:
   fish();
   ~fish();
   void doFishyStuff();
   void setScared(bool value);

  private:
   void swim();
   bool eat();
   bool die();

   float speed;
   float length;
   bool scared;
}

void build(fish* fishes[], int num) {
   fishes[num] = new fish();
   if (dolphinsLeaveYet)
      fishes[num]->setScared(false);
   else
      fishes[num]->setScared(true);
}

int main(int argc, char **argv) {
   int i, n;
   if (argc > 1)
      int n = 100;
   else
```

```
      return 1;
   fish *fishes[n];
   for (i = 0; i < n; i++) {
      build(fishes, i);
      if (i >= 42)
         dolphinsLeaveYet = true;
   }
}
```

For the following variables, give the name, address, type, value, lifetime, scope, and state if the memory needed for the variables if allocated during compile time or during runtime:

(a) `n`

(b) `num`

(c) `fishes[]`

(d) `dolphinsLeaveYet`

Exercise 3 Consider the following Python program:

```
x = 1
y = 3
z = 5
def sub1():
    a = 7
    y = 9
    z = 11
def sub2():
    global x
    a = 13
    x = 15
    w = 17
    def sub3():
        nonlocal a
        a = 19
        b = 21
        z = 23
```

Assume static scoping. List all variables, along with the program units where they are declared, that are visible in the bodies of

(a) `sub1`,

(b) `sub2`, and

(c) `sub3`.

[Hint: You can verify your answer by running the program with Python 3 using `dir()`, `globals()`, and `locals()`]

Exercise 4 Consider the following code:

```
const int b = 5;
int foo()
{
   int a = b + 5;
   return a;
}

int bar()
{
   int b = 2;
   return foo();
}

int main()
{
   foo();
   bar();
   return 0;
}
```

What do `foo()` and `bar()` return when using

(a) static scoping, and

(b) dynamic scoping.