

Warning: These notes are not complete, it is a Skelton that will be modified/add-to in the class. If you want to us them for studying, either attend the class or get the completed notes from someone who did

CSE2301

Dynamic Memory Allocation and Structs

These slides are based on slides by Prof. Wolfgang Stuerzlinger at York University

Pointers to Functions

- Function Declaration
- `returnType f_name (parameterTypes);`
- For example,
- `int mysum(int, int)`
- `mysum` is a function that takes two integers and return an integer.

Pointers to Functions

- We can declare a variable of type pointer to a function.
- That variable is pointing to the function, and can be used the same way we use pointers.
- Useful when we want to pass a function is an argument to another function.
- For example, a function that perform sorting, that implies comparison. We can tell the sorting function to use a specific comparison function by passing a pointer to it.

Pointer to Functions

► Declaration:

```
returnType (*varName)(parameterTypes);
```

► Examples:

```
int (*f)(int, float); A pointer to a function that  
takes one Integer and one float and returns int
```

```
int *(*f)(int, float); A pointer to a function that  
takes one int and one float, and returns a pointer  
to int
```

Example

```
#include <stdio.h>
int f1(int, int, int (*fp)());
int add(int, int);
main() {
    int i,j,k,l; int x,y;
    int z;
    int (*fptr)(int, int);
    fptr=add;
    scanf("%d%d", &x, &y);
    z=f1(x, y, fptr);
    printf("%d\n", z);
}

int add(int i, int j) {
    return(i+j);
}

int f1(int x, int y, int (*ptr)()) {
    return(x*x + y*y + (*ptr)(x,y));
}
```

Example

```
#include <stdio.h>
int f1(int, int, int (*fp)());
int add(int, int);
main() {
    int i,j,k,l; int x,y;
    int z;
    int (*fptr)(int, int);
    fptr=add;
    scanf("%d%d", &x, &y);
    z=f1(x, y, fptr);
    printf("%d\n", z);
}

int add(int i, int j) {
    return(i+j);
}

int f1(int x, int y, int (*ptr)()) {
    return(x*x + y*y + (*ptr)(x,y));
}
```

Example

```
#include <stdio.h>
int f1(int, int, int (*fp)());
int add(int, int);
main() {
    int i,j,k,l; int x,y;
    int z;
    int (*fptr)(int, int);
    fptr=add;
    scanf("%d%d", &x, &y);
    z=f1(x,y, fptr);
    printf("%d\n", z);
}

int add(int i, int j)
{
    return(i+j);
}

int f1(int x, int y,
int (*ptr)()) {
    return(x*x + y*y +
(*ptr)(x,y));
}
```
