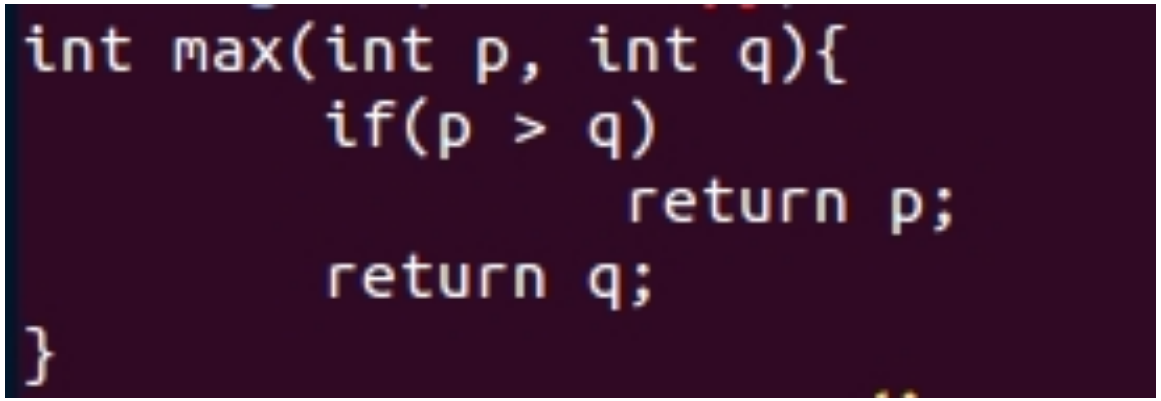
C1扩展实验

PB13011054　陈远昭

P7扩展

- 函数参数（SimpleDeclList）
- 函数返回值
- break/continue（记录循环的块）



```
define i32 @max(i32 %p, i32 %q) {
entry:
  %p1 = alloca i32
  store i32 %p, i32* %p1
  %q2 = alloca i32
  store i32 %q, i32* %q2
  %p3 = load i32* %p1
  %q4 = load i32* %q2
  %gttmp = icmp sgt i32 %p3, %q4
  br i1 %gttmp, label %then, label %else

then:                                             ; preds = %entry
  %p5 = load i32* %p1
  ret i32 %p5
  br label %ifcont

else:                                             ; preds = %entry
  br label %ifcont

ifcont:                                           ; preds = %else, %then
  %q6 = load i32* %q2
  ret i32 %q6
}
```

增加类型

- float
- 指针
- 结构体

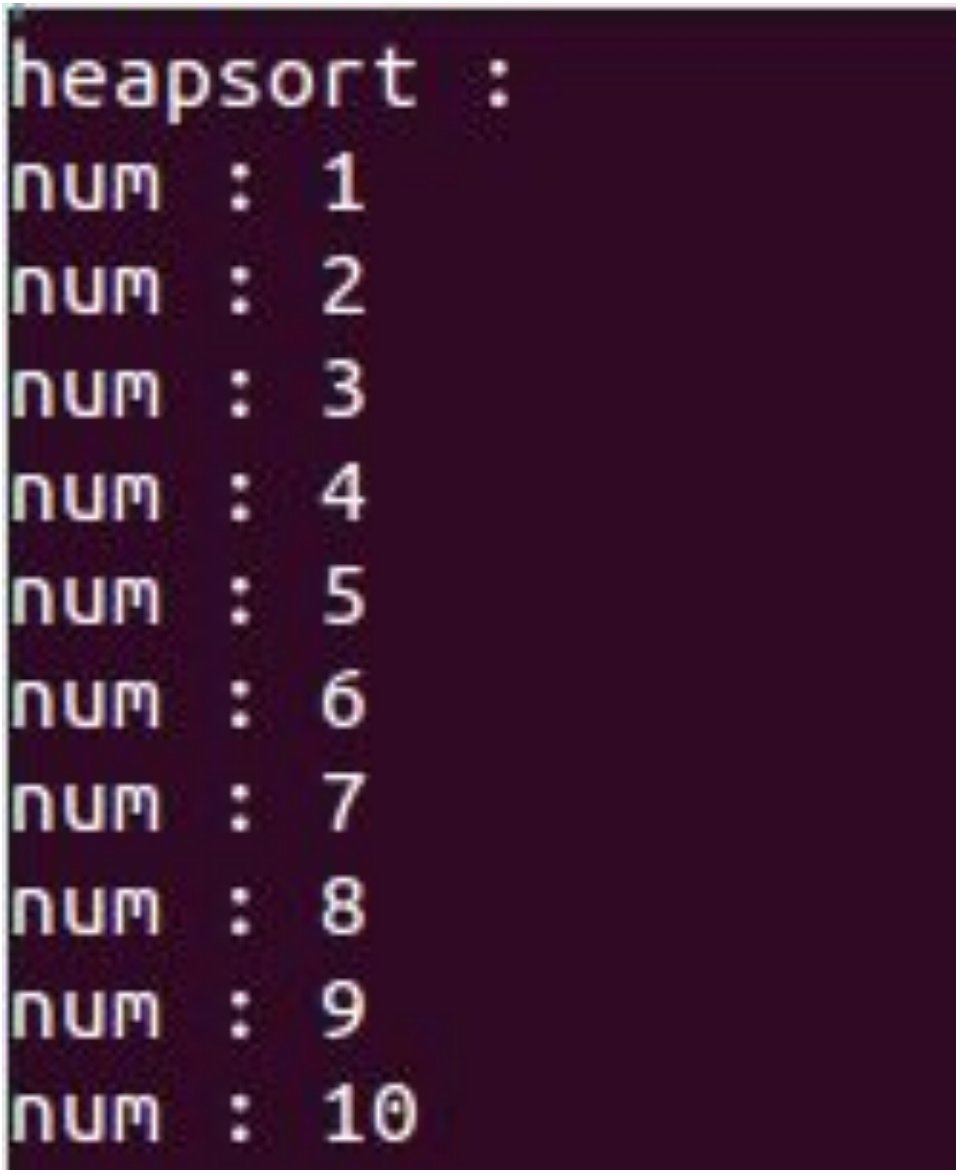
```
➜ P8 git:(finalcopy) ✗ ./run.sh
psrse done
; ModuleID = 'my cool jit'

define i32 @main() {
entry:
  %p = alloca i32*
  %q = alloca i32*
  %q1 = load i32** %q
  store i32* %q1, i32** %p
  %0 = load i32** %p
  %1 = load i32* %0
  %a = alloca i32
  store i32 %1, i32* %a
}
➜ P8 git:(finalcopy) ✗ cat test/t1.c1
int main(){
        int *p, *q;
        p = q;
        int a =*p;
}
```

```
%A = type { i32, double* }

@0 = private constant %A { i32 2, double* null }

define i32 @main() {
entry:
  %lp = alloca %A
  %lq = alloca %A
  %cast = bitcast %A* %lp to i8*
  call void @llvmmemcpy(i8* %cast, i8* bitcast (%A* @0 to i8*), i64 12, i32 8, i1 false)
  %lp1 = load %A* %lp
  store %A %lp1, %A* %lq
}
➜ P8 git:(finalcopy) ✗cat test/t1.c1
struct A{
        int num;
        float *fp;
};

int main(){
        struct A lp;
        struct A lq;
        lp = { 2, 0 };
        lq = lp;
}
```

example 1

- 堆排序（待排序数组是全局变量）

- H[10] = {3, 1, 5, 7, 2, 4, 9, 6, 10, 8} ;



example2

- 简单的链表

- struct LinkList{
- 	int num;
- 	struct LinkList *next;
- };

- 利用链表的结构顺序访问其中的元素