

08

장

서브루틴과 제어 추상화

편집자주: 원서의 문제 번호가 잘못되어 8.34부터 시작하고 있으나 한국어판에서는 8.33부터로 바로 잡습니다.

8.8 연습문제

8.33 MIPS상의 매개변수가 스택 내 위치를 가져야 할 필요가 있을 이유를 정교화하라. 그것의 수명 동안 한 레지스터 내에 매개변수를 저장하는 것을 만족시키지 않을 수 있는 모든 경우를 살펴보라.

8.34 C 라이브러리의 대부분의 버전은 현재 스택 프레임 내에 동적으로 공간을 할당하는 함수, `alloca`를 포함한다.¹ 그것은 스택 내에 공간을 할당하는 일반적인 `malloc`에 비해 두 가지 이점을 갖는다. 이는 보통 매우 빠르고 현재 서브루틴이 복귀할 때 할당한 공간을 자동적으로 반환한다. 프로그래머가 해제가 발생하기를 원한다고 가정한다면 명시적인 `free` 연산을 넘길 수 있는 것이 편리하다. ©(심화학습에 있는) 8.2.2절에서 설명한 MIPSpro 호출 규정과 관련해 `alloca`를 어떻게 구현하겠는가?

8.35 (7.4.2절에서 설명한) 정교화 시점까지 경계를 알지 못하는 배열을 수용하기 위해 ©(심화학습에 있는) 그림 8.9와 ©(심화학습에 있는) 8.2.2절의 규정을 확정하는 방법을 설명하라. 별개의 스택과 프레임 포인터의 사용에 대해 이는 어떤 결과를 가지겠는가?

주1. 불행히도 `alloca`는 POSIX 규격이 아니고 구현에 따라 의미와 세부 인터페이스도 매우 다르다. 이식 가능한 프로그램은 이 루틴을 피하는 것이 현명하다.

8.36 MIPSpro와 gpc 사례 연구 둘 모두에서 인자를 인자 저장 공간에 첫 인자를 꼭대기에 놓으면서 “역” 순서로 위치시킨다. 이것이 중요한 이유를 설명하라 (힌트: 8.3.3절에서 논의한 가변 개수의 인자를 갖는 서브루틴에 대해 생각해보라).

8.37 스택 내 코드를 실행하지 못하게 하는 기계에서 매개변수로 중첩된 서브루틴을 어떻게 구현하겠는가? 단순한 코드 주소를 전달할 수 있거나? 혹은 클로저를 실행 시간에 해석하게 요구하겠는가?

8.38 Sparc상에서 setjmp와 longjmp를 어떻게 구현할 수 있을까?

8.39 (C)(심화학습에 있는) 예 8.67에 이어서 다음의 호출은 실행 시간 예외를 발생시킬 것이다.

```
csNames.consider (null);
```

null 인자를 받게 String.compareTo를 설계하지 않았기 때문이다.

(a) Chooser<T> 인터페이스에서 술어 `public Boolean valid(T a);`를 포함하고 적절한 `consider`를 변경해 이 가능성으로부터 지키기 위해 (C)(심화 학습에 있는) 그림 8.12를 변경하라. 클래스 CaseSensitive를 적절히 변경하라.

(b) (C)(심화학습에 있는) 그림 8.14와 (C)(심화학습에 있는) 예 8.74의 C# Arbiter에 비슷한 변경을 가하는 방법을 제안하라. Better와 Valid 모두를 필요로 할 때 하한을 어떻게 다뤄야 하는가?

8.40 (a) 비교 루틴이 (C)(심화학습에 있는) 그림 8.11의 chooser을 생각나게 하는 명시적인 제네릭 매개변수이게 하기 위해 연습문제 8.22에 대한 답을 변경하라.

(b) 비교 루틴이 sort에 추가적인 매개변수인 대안적인 해결책을 제시하라.

8.41 그림 6.5의 코드에 이어 반복자의 단일 스택 구현을 가정하면 완전한 3 수준(6개의 노드) 이진 트리에 속한 모든 노드에 비해 반복하는 for 루프의 스택의 내용을 추적하라.

8.42 자바, C#, 파이썬에서 이진 트리에 대한 전위 반복자를 만들어라. `true` 반복자나 트리 노드의 명시적인 스택을 사용하지 말라. 오히려 중첩된 반복자 객체를 필요할 때마다 생성하라. `true` 전위 반복자와 동일한 반복자 객체를 만들고 있다면 C# 컴파일러로 그들을 서로 연결할 수 있다.

8.43 ©(심화학습에 있는) 8.6.4절의 교통량 시뮬레이션에서 부정확함의 한 원인은 교통 신호에서 시간적 조절을 처리해야 하는 것에 있다. 동서 방향이 현재 녹색등이지만 대기 자동차의 큐가 비었으면 `signal` 동시 실행 루틴은 `current_time + EW_duration`까지 잠들 것이다. 동시 실행 루틴이 다시 깨어나기 전에 자동차가 도착하면 그것은 기다릴 필요가 없다. 이 문제를 어떻게 고칠 것인지를 논하라.