

Part 4 프로그래밍 언어 활용

Chapter 1 프로그래밍 언어 기초

Section 1 프로그램 언어와 번역

1. Linker는 번역된 프로그램 들은 연결해주는 프로그램
2. Preprocessor - 번역되기 전 다양한 기능 처리
3. 실행, 시작 → 결과출력
4. 메트로 프로세서 → 정의 인식, 정의 저장, 호출 인식
5. C언어, Cobol, Pascal, Java ⇒ 컴파일 언어
6. 전처리기 할당 - 프로그래머 연결 - 프로그래머
재배치 - 어셈블러 작성 - 코더

Section 2 자료 객체

1. 새로운 언어일수록 예약어 수가 많음
2. 포인터 자료형 → 고급언어
3. 포인터: 다른 메모리 공간의 주소를 포함하는 메모리 위치로 나타내는 객체
4. 바인딩: 각종 값들이 객체로써 레이블 변경할 수 없는 구조상태가 되는 것

5. 번역 시간 - 컴파일, 링크, 로드 시간

6. 정적 바인딩 - 번역, 언어 구현 시간 / 동적 \rightarrow 실행 시간

Chd 예상 문제

1. ② 비선형성 \rightarrow 지능 언어

2. ②

3. ② \checkmark 블록 프로그램: 기계어로 번역된 프로그램

4. ④ \checkmark 기계어만 번역과정이 없다.

5. ④

6. ③

7. ④

8. ③

9. ①

10. ④

11. ①

12. ③

13. ㉓ ✓ SLL: 사용자 프로그램이 실행되기전, DLL: 실행된 후

14. ㉔

15. ㉔

16. ㉔ ✓ 링커: 여러개의 목적코드를 코드 모듈로 생성함

17. ㉓

18. ㉔

19. ㉔ 컴파일러: 어휘 → 구문 → 의미 → 중간코드 → 코드최적화 → 생성

20. ㉔

21. ㉔

22. ㉔ 원시 프로그램 → 컴파일러 → 목적 프로그램 → 링커 → 로더

23. ㉓

24. ㉔ ✓ 정적인 조직 + 동적인 조직

25. ㉓

26. ㉔

27. ㉓