



printf

Because putnbr and putstr aren't enough

Summary: 이 프로젝트는 꽤 간단합니다. 당신은 `printf` 함수를 다시 코드화할 것입니다. 바라건대 미래의 프로젝트에서 *Cheater*로 플래그 찍힐 염려 없이 그것을 재사용할 수 있길 바랍니다. 당신은 주로 *variadic arguments*를 어떻게 사용하는지 배우게 될 것입니다.

- * 이 파일은 프로젝트 수행을 좀 더 빠르고 편하게 하기 위해 만들어진 파일입니다.
- * 개인적으로 보기위해 만들어진 파일이므로 오역이 있을 수 있습니다.
- * 영어버전과 함께 보시는 걸 추천 드립니다.

Contents

I	Introduction	2
II	Common Instructions	3
III	Mandatory part	4
IV	Bonus part	5

Chapter I

Introduction

C에서 printf 함수의 다양한 기능은 우리에게 프로그래밍에 있어서 훌륭한 연습이 됩니다. 이 프로젝트는 중간 정도의 난이도입니다. 그것은 당신이 C에서 variadic한 함수들을 발견할 수 있게 해줄 것입니다. 성공적인 ft_printf를 위한 키는 잘 짜여지고 확장 가능한 코드입니다.

Chapter II

Common Instructions

- 프로젝트는 Norm 규칙에 맞춰 작성되어야 합니다. 보너스 파일/함수가 있는 경우, 해당 파일/함수들은 norm 검사에 포함되며, norm error가 있을 시, 0점을 받게 될 것입니다.
- 함수들은 정의되지 않은 행동들과는 별개로 예기치 않게 중단되어서는 안 됩니다.(예를 들어, segmentation fault, bus error, double free 등.)
- 필요한 경우 heap에 할당된 모든 메모리 공간은 적절하게 해제되어야 합니다. 메모리 누수는 용납되지 않을 것입니다.
- 그 과제가 필요하다면, Makefile을 제출해야 합니다. 그것은 -Wall -Wextra -Werror 플래그를 지정하여 컴파일할 것입니다. 그리고 Makefile은 relink 되어서는 안 됩니다.
- Makefile은 최소한 \$(NAME), all, clean, fclean, re를 포함해야 합니다.
- 프로젝트에 보너스를 제출하려면, Makefile에 보너스 규칙을 포함해야 합니다. 이보너스 규칙은 프로젝트의 메인 부분에서 금지되었던 모든 다양한 헤더, 라이브러리, 또는 함수들은 추가해야 할 것입니다. 보너스는 반드시 _bonus.{c/h}라는 다른 파일에 있어야 합니다. 의무적으로 해야 될 파트와 보너스 파트는 별도로 평가될 것입니다.
- 프로젝트에서 당신의 libft를 허용한다면, 소스들과 그것과 연관된 Makefile을 연관된 Makefile과 함께 libft폴더에 복사해야 합니다. 프로젝트의 Makefile은 Makefile을 사용하여 라이브러리를 컴파일한 다음, 프로젝트를 컴파일해야 합니다.
- 이 과제를 제출하지 않고 등급이 매겨지지 않을지라도, 우리는 당신의 프로젝트를 위한 테스트 프로그램을 만들 것을 권장합니다. 그것은 너의 work와 peer's work를 쉽게 테스트할 기회를 제공할 것입니다. 너는 defence하는 동안 이 테스트 프로그램들이 특히 유용하다는 것을 알게 될 것입니다. 사실, defence하는 동안, 너는 너의 테스트 프로그램과 평가 받는 동료의 테스트 프로그램들을 자유롭게 사용할 수 있을 것입니다.
- 할당된 git 저장소에 과제를 제출하세요. 오직 git 저장소에 있는 과제물만 등급이 매겨질 것입니다. 만약 너의 과제를 평가받는데 Deepthought가 지정된다면, 그것은 동료평가 이후에 이루어질 것입니다. 만약 Deepthought 평가 중에 오류가 발생한다면, 그 즉시 평가는 중지될 것입니다.

Chapter III

Mandatory part

Program name	libftprintf.a
Turn in files	*.c, */*.c, *.h, */*.h, Makefile
Makefile	all, clean, fclean, re, bonus
External functs.	malloc, free, write, va_start, va_arg, va_copy, va_end
Libft authorized	yes
Description	Write a library that contains ft_printf, a function that will mimic the real printf

- ft_printf의 프로토타입은 `int ft_printf(const char *, ...)`; 이어야 합니다.
- 당신은 libc의 printf 함수를 recode해야 합니다.
- 원래의 printf 함수처럼 buffer 관리를 해서는 안 됩니다.
- 다음과 같은 conversions를 관리 할 것입니다: cspdiuxX%
- 모든 conversions에서 '-0.*' 플래그와 minimum field width의 조합을 관리 할 것입니다.
- 원래의 printf 함수와 비교될 것입니다.



man 3 printf / man 3 stdarg

Chapter IV

Bonus part

- 만약 Mandatory part가 완벽하지 않다면, 보너스에 대해서는 생각조차 안 할 것입니다.
- 모든 보너스를 수행할 필요는 없습니다.
- 다음 conversions 중 하나 이상을 관리하십시오 : nfge
- 다음 flags 중 하나 이상을 관리하십시오 : l ll h hh
- 다음 모든 flags 를 관리하십시오 : '# +' (yes, one of them is a space)