컴퓨터학부 20162518 최승서

1. **개요**
   1. **prime .sh**

prime.sh는 두개의 정수를 입력 받은 후 두 정수 사이의 모든 소수의 합을 구하는 쉘 스크립트 프로그램이다.

2개의 숫자를 받은 후 1번째 숫자부터 2번째 숫자까지 1씩 증가하는 반복문을 수행하여 증가하는 숫자별로 해당 숫자가 소수인지 판별하는 로직을 통하여 소수로 판별된 숫자들을 더하여 프로그램을 구현하였다.

해당 숫자가 소수인지 판별하는 로직은 간단하게 가장 작은 소수인 2부터 자기 자신 전까지 나누어 떨어지는 숫자가 있는지 확인하고 나누어 떨어지는 숫자가 존재하면 해당 숫자는 합산에서 제외하는 로직이다.

* 1. **day.sh**

day.sh는 월과 일을 입력 받은 후 해당 날짜가 무슨 요일인지 출력하는 쉘 스크립트 프로그램이다.

먼저 각 월별로 며칠까지 있는지 배열 monthDay를 선언하여 저장을 해놓고 2021년도의 시작 요일인 금요일부터 모든 요일을 문자열로 DAY변수에 저장을 한다.

달과 월을 입력 받은 후 입력한 날짜가 실제 존재하는 날짜인지 확인하고 실제 존재하는 날짜일 경우에만 2021년 1월 1일부터 입력한 날짜까지 모든 일수를 더하고 %7을 하고 나머지를 확인하여 요일을 출력하는 로직이다.

1. **소스코드와 주석**

**2-1. prime.sh**

#!/bin/bash

echo "INPUT FIRST NUMBER : "

read num1

echo "INPUT SECOND NUMBER : "

read num2

declare -i i j

declare -i cnt

declare -i sum

sum=0

for((i=$num1; i<=$num2; i++))

do

cnt=0

for((j=2; j<$i; j++))

do

if(($i%$j==0))

then

cnt=$cnt+1

fi

done

if((cnt==0))

then

if(($i!=1))

then

sum=$sum+$i

fi

fi

done

echo "SUM OF PRIME NUMBERS($num1 to $num2) : $sum"

**2-2. day.sh**

#!/bin/bash

declare -i monthDay

declare -i month

declare -i day

monthDay=(31 28 31 30 31 30 31 31 30 31 30 31)

DAY=(금 토 일 월 화 수 목)

echo "INPUT MONTH : "

read month

echo "INPUT DAY : "

read day

if(( ${monthDay[$month-1]} < $day ))

then

echo "잘못된 날짜입니다."

else

echo "$month월 $day일 "

#요일 계산 부분

for((i=0;i<$month-1;i++))

do

day=$day+${monthDay[$i]}

done

let index=$day%7-1

echo "${DAY[$index]}요일"

fi

1. **실행결과**

**3-1. prime.sh**

**텍스트이(가) 표시된 사진

자동 생성된 설명**

**텍스트이(가) 표시된 사진

자동 생성된 설명**

**텍스트이(가) 표시된 사진

자동 생성된 설명**

**3-2. day.sh**

**텍스트이(가) 표시된 사진

자동 생성된 설명**

**텍스트이(가) 표시된 사진

자동 생성된 설명**

**텍스트이(가) 표시된 사진

자동 생성된 설명**

1. **구현을 통해 배운 점**

prime.sh와 day.sh 구현을 통하여 bash shell script에서의 변수 선언(정수형 변수, 리스트 변수), 산술 연산, 비교 연산, if문, for문 등 기본적인 쉘 스크립트 문법으로 간단한 로직의 프로그램을 작성함으로서 쉘 스크립트 문법 이용하여 프로그래밍 하는 방법을 배울 수 있었다.