**2024 Spring OOP Assignment Report**

과제 번호 : 2-2

학번 : 20230642

이름 : 이채영

Povis ID : chyng

**명예서약 (Honor Code)**

나는 이 프로그래밍 과제를 다른 사람의 부적절한 도움 없이 완수하였습니다.

I completed this programming task without the improper help of others.

1. **프로그램 개요**

* 본 프로그램은 16진수 문자를 읽어 암호문을 해독하는 프로그램이다.
* 프로그램을 실행하고 “..”로 끝나는 암호문을 입력하면, 암호문에 포함된 노이즈의 비율을 출력하고, 암호문에 포함된 16진수 숫자를 모은 문자열을 출력하고 문자들을 두 개씩 묶어 ASCII 코드로 디코딩된 암호문을 출력한다.

1. **프로그램의 구조 및 알고리즘**

* 먼저 원본 암호문을 입력받는다. 원본 암호문은 string에 저장되고, “..”을 인식하여 암호문의 마지막을 결정한다. “..”까지 암호문의 원본에 저장되면 그것을 바탕으로 calculate\_noise(string text) 함수에서 암호문의 노이즈 비율을 계산한다. 그 다음 암호문을 해독하기 위해 암호문에 포함된 16진수 숫자들을 모아 hexadecimal\_text라는 이름의 string에 저장하고(hexadecimal(string text) 함수 사용), 이를 출력한다. 그리고 이 string의 문자들을 두 개씩 묶어 ASCII 코드로 디코딩해 해독된 암호문을 출력한다.(decryption(string text) 함수 이용)
* 각 함수 설명
  + calculate\_noise(string text): 노이즈 계산, 노이즈 비율 리턴. 노이즈 문자의 개수를 noise\_cnt 변수에 저장한다. 그렇게 되면 노이즈 비율 식이 100 x (noise\_cnt) / (text.size()) 가 되므로 이 값을 리턴한다.
  + hexadecimal(string text): 원본 암호문에서 16진수 문자를 추출하여 따로 string에 저장. <cctype> 헤더파일에 내장된 함수 isxdigit(char), isdigit(ch), isupper(ch) 를 이용해 16진수 문자 중 숫자(0~9)와 대문자(A~F)만 추출하여 따로 문자열에 저장한다. 그리고 따로 저장한 문자열을 리턴한다.
  + decryption(string text): 16진수 문자만 저장한 문자열을 바탕으로 암호문을 해독. 16진수 문자의 개수가 홀수이거나 16진수 숫자가 없는 경우는 예외 처리한다. 16진수 문자를 두 개씩 묶고, <string> 헤더 파일에 내장된 함수인 stoi(const std::string& str, std::size\_t\* pos = nullptr, int base) 함수에서 base 자리에 16을 넣어 16진수를 10진수로 변환한다. 변환한 숫자가 아스키 코드 범위를 벗어날 경우(126 초과) 예외 처리를 해준다. 그리고 아스키 코드로 디코딩한 후 디코딩된 문자를 decrypted\_text라는 문자열에 추가한다. 모두 디코딩한 후에는 decrypted\_text 문자열을 출력한다.

1. **토론 및 개선**
   * + 이번 과제를 통해 c++에서의 Loop, branch문의 사용을 익힐 수 있었다. 또한 <cctype> 헤더 파일을 통해 문자열에서 특정 조건의 문자들을 추출할 수 있음을 알 수 있었다.
     + 이번 과제에서 loop, branch문이 많이 사용되었는데, loop와 branch문이 실행되는 횟수를 줄이면 프로그램의 실행 속도를 높일 수 있다.
       1. 디코딩을 하는 void 함수 decryption(string text)에서 예외 사항이 발생하면 바로 리턴하도록 하여 프로그램의 실행 속도를 높이고자 하였다.
       2. 원본 문자열에서 “..”이 발견되면 문자열의 마지막으로 인식하는 부분에서, text.back() 함수가 정상적으로 작동되려면 text.size() >= 1이여야 한다는 조건을 앞에서 통과해야 한다. (빈 문자열에서 text.back()을 실행하면 오류 발생) 이를 구현할 때 논리 연산자를 사용해 if(ch == '.' && cypertext.size() >= 1 && cypertext.back() == '.') 와 같이 코딩을 하면 실행 횟수를 가장 줄일 수 있어 효과적으로 프로그램이 작동할 수 있다.
2. **참고 문헌**

* <https://en.cppreference.com/w/cpp/string/basic_string/stol>
  + 헤더파일 <string>의 내장 함수 stoi()를 사용하여 16진수를 10진수로 변환하기 위해 위의 사이트를 참고하였다.