プログラミングC++　第6回クイズ

学生番号：s1270151　 氏名：上田健登

以下のstring型の文字列xmlと構造体Tagがある．このとき「＜」から「＞」までの間の部分（タグ）を構造体のtagに保存し（例えばnoun），その後の「＜／」までの部分（例えばI）を構造体のelementに分解して，全体を変数wordsに保存するプログラムを書け．

struct Tag {

string tag;

string element;

}

std::string xml = “<noun>I</noun> <verb>talk</verb> <adverb>about</adverb> <noun>running</noun>”;

Std::list<Tag> words;

We have the above struct Tag, string object of named xml, and lists. Write a c++ source code for

(1) Extract a string between “<” and “>” and store it in tag in the struct.

(2) Extract a string between “>” and “<” and store it in element in the struct..

(3) Put the struct to lists.

解答欄 / Answer

#include <iostream>

#include <string>

#include <list>

#include <algorithm>

struct Tag {

std::string tag;

std::string element;

};

int main(){

std::string xml = "<noun>I</noun> <verb>talk</verb> <adverb>about</adverb> <noun>running</noun>";

std::string open\_tag\_begin = "<";

std::string open\_tag\_end = ">";

std::string close\_tag\_begin = "</";

std::string close\_tag\_end = ">";

std::list<Tag> words;

std::string::iterator head = xml.begin();

const std::string::iterator end = xml.end();

std::string::iterator i, j;

while(head != end){

Tag word;

i = std::search(head, end, open\_tag\_begin.begin(), open\_tag\_begin.end());

head = i + 1;

j = std::search(head, end, open\_tag\_end.begin(), open\_tag\_end.end());

head = j + 1;

std::copy(i + 1, j, std::back\_inserter(word.tag));

i = std::search(head, end, close\_tag\_begin.begin(), close\_tag\_begin.end());

head = i + 2;

std::copy(j + 1, i, std::back\_inserter(word.element));

words.push\_back(word);

j = std::search(head, end, close\_tag\_end.begin(), close\_tag\_end.end());

head = j + 1;

}

for (std::list<Tag>::iterator it = words.begin(); it != words.end(); it++) {

std::cout << it -> tag << ": " << it -> element << '\n';

}

return 0;

}