Practical 1(b): Program to determine uniquely decodable codes.

```
#include<iostream>
#include<vector>
using namespace std;
vector<string> v;
void printlist()
{
	cout<<"List : ";
		for(int k=0;k<v.size();k++)
			cout<<v[k]<<"   ";
		cout<<endl<<endl;
}
void checkthis(string temp,int len)    //temp = latest added string len=n
{
	for(int i=0;i<len;i++)
	{
		int upto=v[i].size();
		if(v[i].size() < temp.length() && v[i]==temp.substr(0,upto))
		{
			string test=temp.substr(upto);
			v.push_back(test);
			cout<<"Added "<<test<<"  in the list because element "<<i+1<<
			" is prefix of element "<<v.size()-1<<endl;
			printlist();
			checkthis(test,len);
		}
	}
}

int main()
{
	int len,decodable=true,lastindex;
	cout<<"How many codes you want to enter : ";cin>>len;
```

```
for(int i=0;i<len;i++)
{
	string temp;
	cout<<i+1<<") :";cin>>temp;
	v.push_back(temp);
}
printlist();
for(int i=0;i<v.size();i++)
{
	for(int j=0;j<v.size();j++)
	{
	if(i!=j)
	{
		if(v[i]==v[j] && (i<len || j<len) )
		{
			cout<<"Element "<<i+1<<" and "<<j+1<<" is matching so
			it can not be UDC.";
			decodable=false;
			break;
		}
		else if( v[i].size() < v[j].size() &&
		v[i]==v[j].substr(0,v[i].size()) && (i<len || j<len) &&
		j!=lastindex)
		{
			string temp;
			temp=v[j].substr(v[i].size(),v[j].size());
			lastindex=j;
			cout<<"Added "<<temp<<"  in the list because element
			"<<i+1<<" is prefix of element "<<j+1<<endl;
			v.push_back(temp);
			printlist();
			checkthis(temp,len);
		}
	}
	}
	if(decodable==false)
	break;}
```

```
	if(decodable)
		cout<<endl<<"Uniquely Decodable";
	else
		cout<<endl<<"Not Uniquely Decodable";
}
```

OUTPUT: