Практична робота №2
Завірюха Еліна, МП-21

Завдання 1:

```
#include <iostream>

struct Node
{
int data;
Node *next;
};

Node *create()
{
int n;
Node *head = nullptr;
Node *newNode;

std::cout<<"Enter the number of elements in the list: ";
std::cin>>n;

for (int i = 0; i < n; i++)
{
newNode = new Node;
std::cout<<"Enter the value of element "<<i + 1<<": ";
std::cin>>newNode->data;
newNode->next=head;
head=newNode;
}

return head;
}



void add(Node*& head)
{
int n;
Node* newNode;
std::cout<<"Enter the number of elements to add to the list: ";
std::cin>>n;

for (int i = 0; i < n; i++)
{
newNode = new Node;
std::cout<<"Enter the value of the new element "<<i + 1<<": ";
```

```
std::cin>>newNode->data;
newNode->next = head;
head = newNode;
}
}

void displayList(Node* head)
{
Node* current = head;
while (current != nullptr)
{std::cout<<current->data<< " ";
current=current->next;
}
std::cout << std::endl;
}



int main()
{Node*list1 = create();
std::cout<<"List created using createListFromInput: ";
displayList(list1);

Node*list2 = nullptr;
add(list2);
std::cout<<"List created using addElementsToList: ";
displayList(list2);

while (list1 != nullptr)
{
Node*temp=list1;
list1=list1->next;
delete temp;
}

while (list2 != nullptr)
{Node* temp=list2;
list2=list2->next;
delete temp;
}

return 0;
}
```

Завдання 3:

```
#include <iostream>
```

```
#include <string>
#include <vector>


enum class Amplyua
{
Goalkeeper,
Defender,
Midfielder,
Forward
};


struct FootballPlayer
{
std::string lastName;
Amplyua position;
int age;
int numberOfGames;
int numberOfGoals;
};


FootballPlayer findBest(const std::vector<FootballPlayer>& players)
{
FootballPlayer bestForward;
for (const FootballPlayer&player:players)
    {
    if (player.position == Amplyua::Forward)
        {
        if (bestForward.numberOfGoals < player.numberOfGoals)
            {
                bestForward = player;
            }
        }
    }
return bestForward;
}



void print5Games(const std::vector<FootballPlayer>& players)
{
std::cout << "Football players who played less than 5 games:" << std::endl;
for (const FootballPlayer& player : players)
    {
    if (player.numberOfGames < 5)
```

```
        {
        std::cout<<"Last Name: "<<player.lastName<<std::endl;
        std::cout<<"Position: ";
        switch (player.position)
        {
        case Amplyua::Goalkeeper:
        std::cout<<"Goalkeeper";
        break;
        case Amplyua::Defender:
        std::cout<<"Defender";
        break;
        case Amplyua::Midfielder:
        std::cout<<"Midfielder";
        break;
        case Amplyua::Forward:
        std::cout<<"Forward";
        break;
        }
            std::cout << std::endl;
            std::cout << "Age: " << player.age << std::endl;
            std::cout << "Number of Games: " << player.numberOfGames <<
std::endl;
            std::cout << "Number of Goals: " << player.numberOfGoals <<
std::endl;
            std::cout << "-----------------------" << std::endl;
        }
    }
}




int main() {
std::vector<FootballPlayer> players =
{
    {"Ronaldo", Amplyua::Forward, 36, 10, 7},
    {"Messi", Amplyua::Forward, 34, 12, 8},
    {"Neuer", Amplyua::Goalkeeper, 35, 8, 0},
    {"Ramos", Amplyua::Defender, 34, 11, 2},
    {"Modric", Amplyua::Midfielder, 35, 7, 1},
    {"Kane", Amplyua::Forward, 28, 4, 3}
};



FootballPlayer bestForward=findBest(players);
std::cout<<"Best Forward:"<<std::endl;
std::cout<<"Last Name: "<<bestForward.lastName<<std::endl;
```

```
std::cout<<"Position: Forward"<<std::endl;
std::cout<<"Age: " << bestForward.age<<std::endl;
std::cout<<"Number of Games: " <<
bestForward.numberOfGames<<std::endl;
std::cout<<"Number of Goals: " << bestForward.numberOfGoals<<std::endl;
std::cout << "-----------------------" << std::endl;

print5Games(players);

return 0;
}
```