

concept of human capital is little understood both at the level of public administration, and manufacturing and personal levels. Nationwide for Ukraine the problem of innovative development, at constant conditions of use of its own intellectual resource, applies not only to the scale of the state strategy and all individual and business entities, including businesses large or small.

Analyzing the labor potential of regions of Ukraine we can conclude that the modern labor markets have a negative impact, because this situation has worsened crisis, unemployment, demographic problems and problems of education.

Thus, qualitative changes in the formation of human capital especially its innovative part, requires active state support for the implementation of innovative development of national economy of Ukraine.

*Scientific supervisor: Grytsai S.Y.,
Senior Lecturer*

UDC 004.4:004.45 (043.2)

Orynych L.V.

National Aviation University, Kyiv

MAIN CONCEPTS OF OBJECT-ORIENTED PROGRAMMING

From the early days of programming software developers have tried slash development time and facilitate the programming process, re-use and streamlined components to reduce program support costs. These efforts led to the creation of object-oriented programming and its principles (abstraction, encapsulation, inheritance and polymorphism), which will be mentioned below.

Concept of encapsulation allows to understand in what way the language hides details of internal implementation object and keeps data consistency.

It is the mechanism that binds together code and data, and saves both from outside interference and wrong usage. In a whole, it isolates different parts of code from each other and from users. A precise interface regulates the access to that particular code.

In the object-oriented programming a combination of data and operations on it is called class. In terms to encapsulation the class should be considered as a shell so user can see and operate only interface part without details of implementation. Most of the data are usually encapsulated. So, they class a thing, which becomes the 'capsule' for the data and operations.

Abstraction answers the question in what way the language allows to define and use objects which correspond to complex entities of real world.

Concept of abstraction is very simple to understand. Abstraction defines as a development of classes, objects, types in terms of their interfaces and

functionality, instead of their implementation details, which makes it similar with principle of encapsulation. Abstraction denotes a model, a view, or some other focused representation for an actual item. It's the development of a software object to represent an object we can find in the real world.

The principle of inheritance explains in what way the language promotes code reusing. Inheritance is possibility to create derived class from base one with saving properties and behavior of base class and to add new properties and behavior if it's necessary. Sequence of classes related by inheritance is a hierarchy. A hierarchy is widely shared in the real world. It is the process by which one object acquires the properties of another object.

And the last concept, polymorphism, gives an explanation how the language allows to use similar objects in the same way.

Polymorphism provides for software construction (usually subroutines) the property to have multiple forms (implementation). Therefore, polymorphism is an ability to hide many different implementations behind a single interface

Polymorphism means the ability to take more than one form. An operation may exhibit different behaviors in different instances. The behavior depends on the data types used in the operation. In general, polymorphism means "one interface, multiple methods".

There are 2 basic types of polymorphism. Overriding, also called run-time polymorphism, and overloading, is referred to as compile-time polymorphism. This difference is, for method overloading, the compiler determines which method will be executed, and this decision is made when the code gets compiled. Which method will be used for method overriding is determined at runtime based on the dynamic type of an object.

*Scientific supervisor: Denysenko N.G.,
Senior Lecturer*

UDC 378.14:111 (043.2)

Ostapchuk V.A.

National Aviation University, Kyiv

PROBLEMS OF LEARNING THE ENGLISH LANGUAGE FOR SPECIFIC PURPOSES

In the modern world, in addition to profession knowledge of a specialist should know foreign language it brings a competitive advantage to her/him. So we can say that the main task of as a student is to acquire communication skills in English.

While learning English for professional purpose a student especially majoring at technical or economic subjects is compressed by many challenges: different