Barbara Liskov

Group N

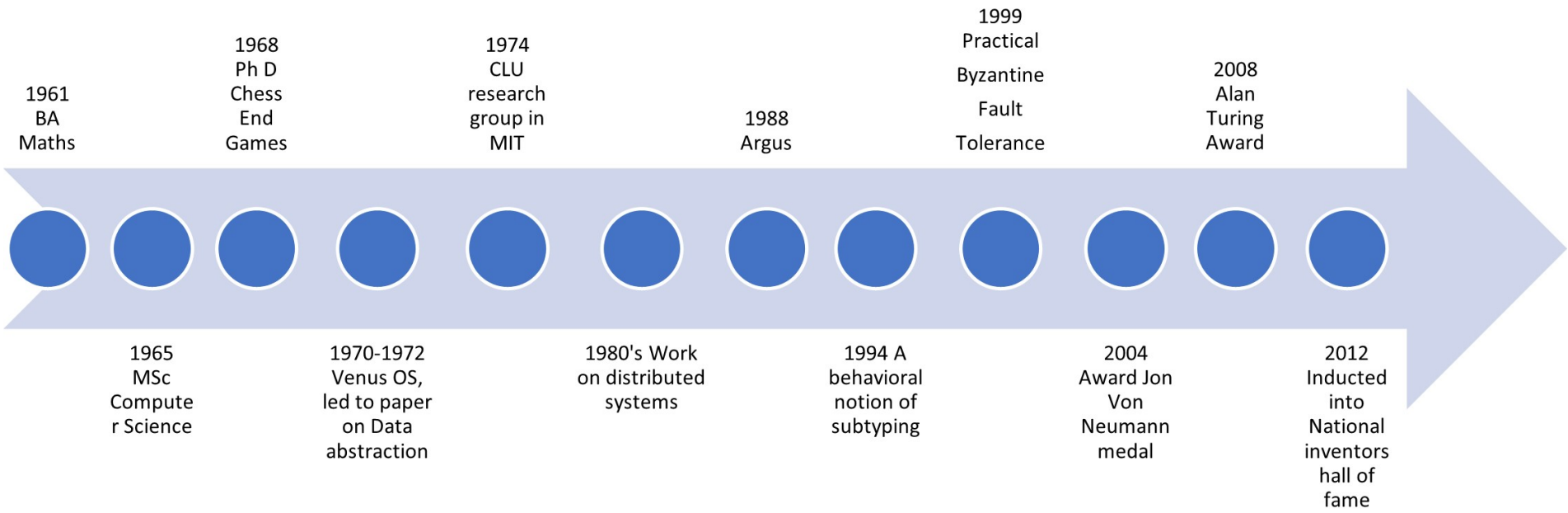
Grace Keane
Shirin Nagle

Brief Description

Important Details

- Computer Scientist
- Born: November 7th, 1939, California
- One of the first women in the USA to be awarded a PhD in Computer Science
- She is one of the world's leading authorities on computer language and system design
- She won numerous awards as well as the Turing award in 2009
- Since 1966, 70 computer scientists have won the Turing Award. Only 3 have been women. Therefore this is an amazing achievement
- Created the Liskov's substitution principle

Barbara Liskov Timeline



Turing Award Citation

Barbara Liskov was awarded the Alan Turing award in 2008 for contributions "of lasting and major technical importance to the computer field".

Award for contributions to practical and theoretical foundations of programming language and system design, especially related to data abstraction, fault tolerance, and distributed computing[1].

Selected work

- Data Abstraction.
- CLU Programming Language.
- Liskov Substitution.
- Distributed Systems.
- Practical Byzantine Fault Tolerance.

Data Abstraction

Because systems of any size can always be expected to be subject to changes in requirements, the project goal is to produce not only reliable software, but readable software which is relatively easy to modify and maintain[2].

What we desire from an abstraction is a mechanism which permits the expression of relevant details and the suppression of irrelevant details. In the case of programming, the use which may be made of an abstraction is relevant; the way in which the abstraction is implemented is irrelevant[3].

This work resulted in CLU a programming language which is a predecessor to many OOP languages.

Her contributions have influenced advanced system developments and set a standard for clarity and usefulness

CLU Programming Language

At MIT she led the design and implementation of the CLU programming language, which emphasized the notions of modular programming, data abstraction, and polymorphism. These concepts are a foundation of object-oriented programming used in modern computer languages such as Java and C#.

CLU introduced many features that are used widely now, and is seen as a step in the development of object-oriented programming and was also notable for its use of classes with constructors and methods, but without inheritance [4].

Cluster structure in CLU

```
int_set = cluster is create, member, size, insert, delete, elements

    rep = array[int]

    % implementations of operations go here

    end int_set
```

Figure: Example of Cluster structure[5].

Barbara Liskov's definition

Developed a new notion of sub typing now known as the Liskov Substitution principle.

At a high level, the LSP states that in an object-oriented program, if we substitute a superclass object reference with an object of any of its subclasses, the program should not break [6].

"If for each object o1 of type S there is an object o2 of type T such that for all programs P defined in terms of T, the behavior of P is unchanged when o1 is substituted for o2 then S is a subtype of T"

```
[7]
// Animal super class
public static class Animal {
  public String favoriteFood;
  public Animal(String favoriteFood) {
    this.favoriteFood = favoriteFood;
  }
}
// Subclass
public static class Dog extends Animal {
  public Dog(String favoriteFood) {
    super(favoriteFood);
  }
}

// Subclass
public static class Cat extends Animal {
  public Cat(String favoriteFood) {
    super(favoriteFood);
  }
}
```

[7]

```
1  // Method to give treats
2  public static void GiveTreatTo(Animal animal) {
3    String msg = "You fed the " + animal.getClass().
       getSimpleName() + " some "  + animal.favoriteFood;
4    System.out.println(msg);
5  }
6
7  // Assigning treats to animals
8  // Do not have to create a new method per animal because of
       the LSP principle
9  public static void main(String[] args) {
10   Dog rover = new Dog("bacon");
11   Cat bingo = new Cat("fish");
12
13   GiveTreatTo(rover);
14   GiveTreatTo(bingo);
15 }
16 Command prompt output:
17
18 You gave the Dog some bacon
19 You gave the Cat some fish
```

SOLID

Figure: Solid.

Principle naming

She did not name the Liskov Substitution Principle. Apparently, she received an email in the 90's by somebody asking her whether he got her principle right, surprising her. She had not known that the principle had borne her name for years in the community.

Distributed Systems

In the 1980's remote file systems came into existence. Worked on an algorithm that handled benign failures or crash failures. Designed a replication technique with Brian Oki[9].

One of the potential benefits of distributed systems is their use in providing highly-available services, that is, services that are likely to be up and accessible when needed. Availability is essential to many computer-based services; for example, in airline reservation systems the failure of a single computer can prevent ticket sales for a considerable time, causing a loss of revenue and passenger goodwill. Availability is achieved through replication. By having more than one copy of important information, the service continues to be usable even when some copies are inaccessible, for example, because of a crash of the computer where a copy was stored[9].

Practical Byzantine Fault Tolerance

In the 1990's came up with a new replication algorithm that is able to tolerate Byzantine faults with Miguel Castro[10].

pBFT consensus rounds are called views and are broken into 4 phases:

1. A client sends a request to the leader node to invoke a service operation.

2. The leading node broadcasts the request to the backup nodes.

3.The nodes execute the request, then send a reply to the client.

4. The client awaits f+1 replies from different nodes with the same result, where f represents the maximum number of potentially faulty nodes[11].

Conclusion

- At 81, Barbara Liskov is still active today contributing to and writing many papers as recently as this year.
- Subsequent work has mainly been in the area of distributed systems.
- Her research has covered many aspects of OS and computation
  - work on object-oriented database systems
  - garbage collection
  - caching, persistence, recovery,
  - security, decentralized information flow
  - modular upgrading of distributed systems, geographic routing
  - fault tolerance and practical Byzantine fault tolerance
- Many of these, deal with situations where a complex system fails in arbitrary ways.
- Liskov developed methods to allow correct operation even when some components are unreliable.

The End

References

[1] Turing Award Citation
https://amturing.acm.org/award$_w$*inners*/*liskov*$_1$108679.*cfm*

[2] A design methodology for reliable software systems, Barbara Liskov, 1972 https://ckrybus.com/static/papers/liskov1972.pdf

[3] Programming With Abstract Data Types - B.Liskov 1974
http://web.cs.iastate.edu/ hridesh/teaching/362/07/01/papers/p50-liskov.pdf

[4] CLU programming language
https://en.wikipedia.org/wiki/CLU$_($*programming*$_l$*anguage*$)$

[5] Example of cluster
https://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.46.9499
rep=rep1&type=pdf

References

[6] The Liskov Substitution Principle Explained
https://reflectoring.io/lsp-explained/

[7] Liskov Substitution Principle in 3 Minutes
https://dev.to/erikwhiting88/liskov-substitution-principle-in-3-minutes-2dc6

[8] SOLID Principle in Programming: Understand With Real Life Examples
https://www.geeksforgeeks.org/solid-principle-in-programming-understand-with-real-life-examples/#:~:text=Liskov's

[9] Viewstamped Replication: A New Primary Copy Method to Support Highly-Available Distributed Systems B. Liskov, B.Oki 1988
http://pmg.csail.mit.edu/papers/vr.pdf

References

[10] PracticalByzantineFaultTolerance M.Castro, B. Liskov 1999
http://pmg.csail.mit.edu/papers/osdi99.pdf

[11] What is practical byzantine fault tolerance
https://crushcrypto.com/what-is-practical-byzantine-fault-tolerance/

Clickable References

- A.M. Turing; Barbara Liskov;
- A history of CLU Barbara Liskov (1992)
- Barbara Liskov Index of publications
- Barbara Liskov CV
- Barbara Liskov home page
- A History of CLU, Barbara Liskov 1992

GitHub link

Organization used to manage our LaTeX presentation progression

GitHub organisation link

Questions?