



**法政大学**  
HOSEI University

# AIとHPC

## Artificial Intelligence and High Performance Computing

数納 広哉

法政大学情報メディア教育研究センター

情報メディア教育研究センターシンポジウム2024

「AIと教育」

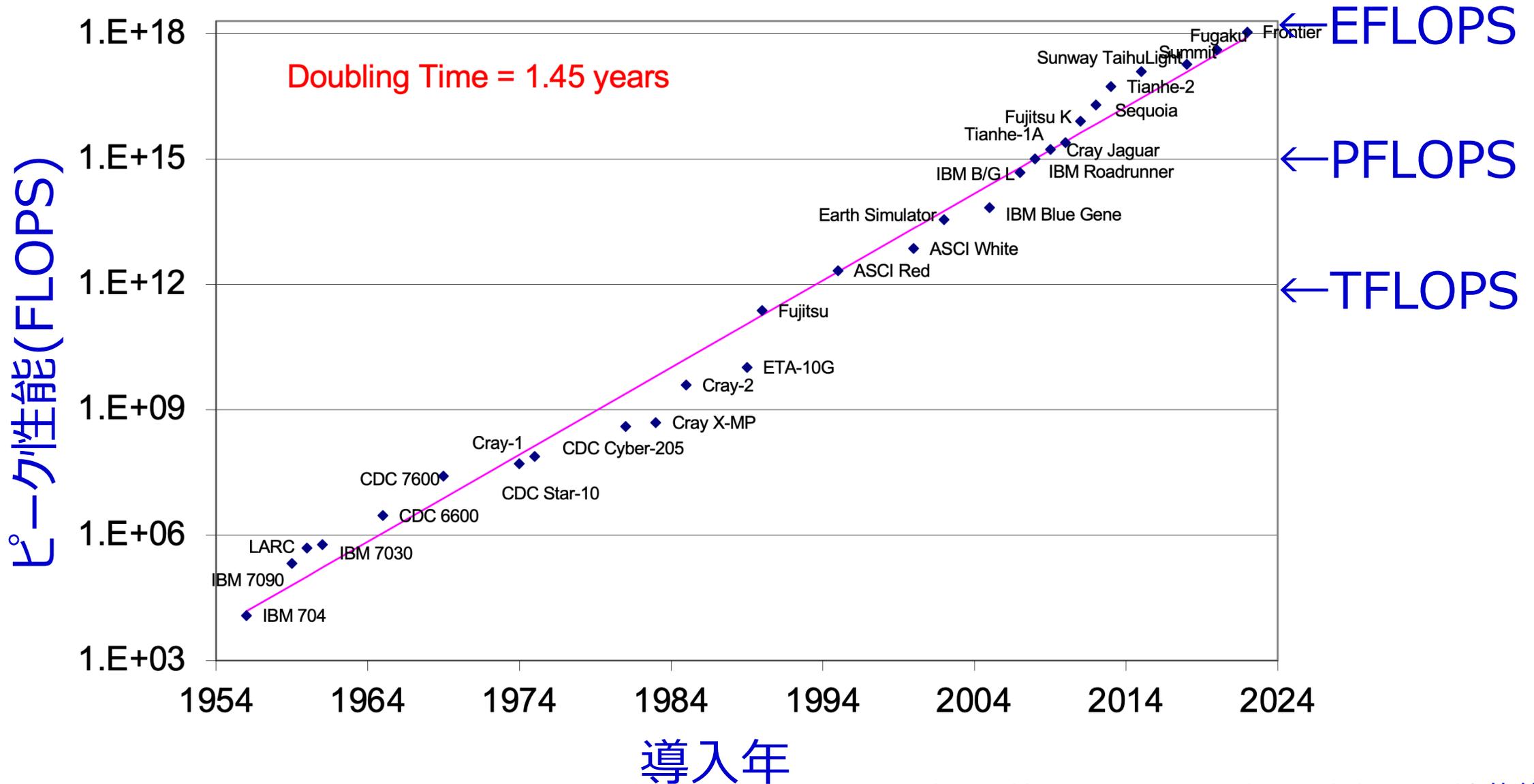
2024年 2月 20日

# 本発表の内容

- 2023年度の研究プロジェクト
  1. lab2022上での数値計算・ディープラーニング(DL)ライブラリの基盤開発
  2. 計算科学シミュレーションソフトウェアの基盤開発
  3. プログラム高速化支援研究
  4. xR技術を活用した教育支援ツールの開発
  5. 計算科学共同研究
- 本発表の内容はプロジェクト1.と3. に関連する：
  - HPC分野の概要・プログラミング技術
  - AI分野の概要・プログラミング技術
  - AIとHPCの融合に向けて

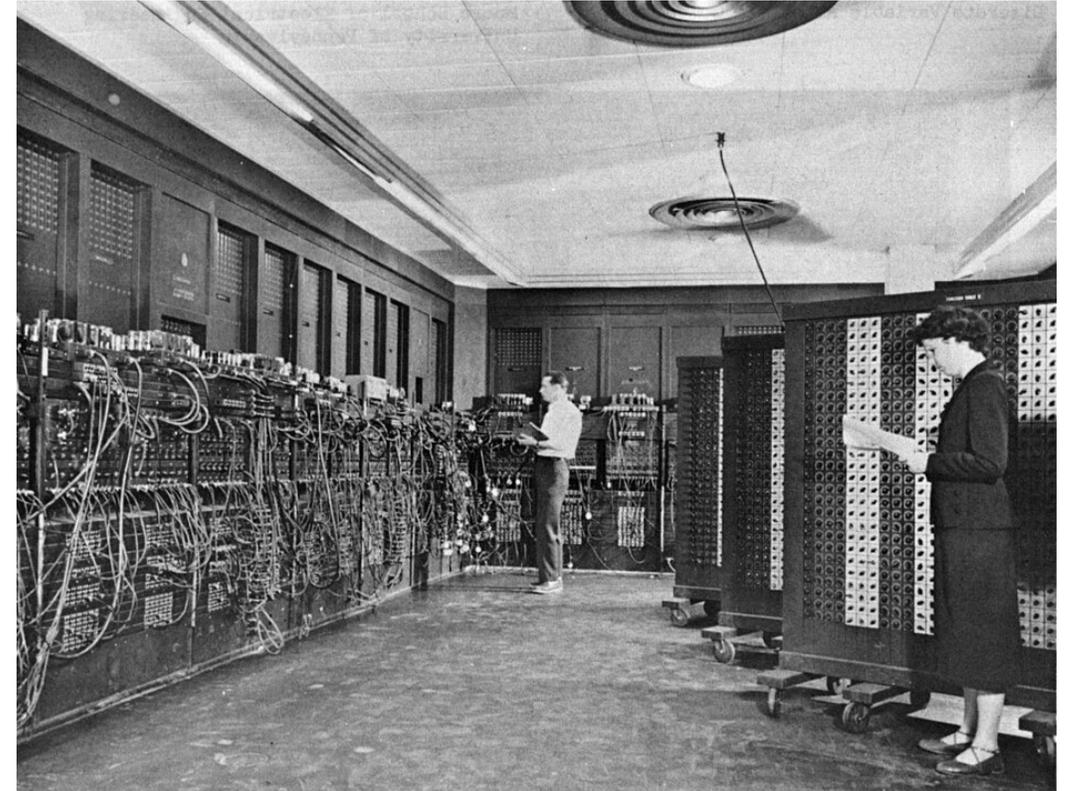
# スーパーコンピュータとHPC

- **スーパーコンピュータ**
  - 最も速いコンピュータを指す。
  - 多くの国で国家が主導して導入している。
- **FLOPS**(フロップス)
  - 処理速度の単位、1秒間の浮動小数点演算回数  
(現在はペタ $10^{15}$ PFLOPS~エクサ $10^{18}$ EFLOPS)
- **HPC**(High Performance Computing, 高性能計算)
  - (おそらく)通常のパソコンの10倍以上の計算能力
- LINPACKベンチマークにより性能評価され、**TOP500**プロジェクトによりランキングが発表される(毎年2回)。



# 初期のスーパーコンピュータ1

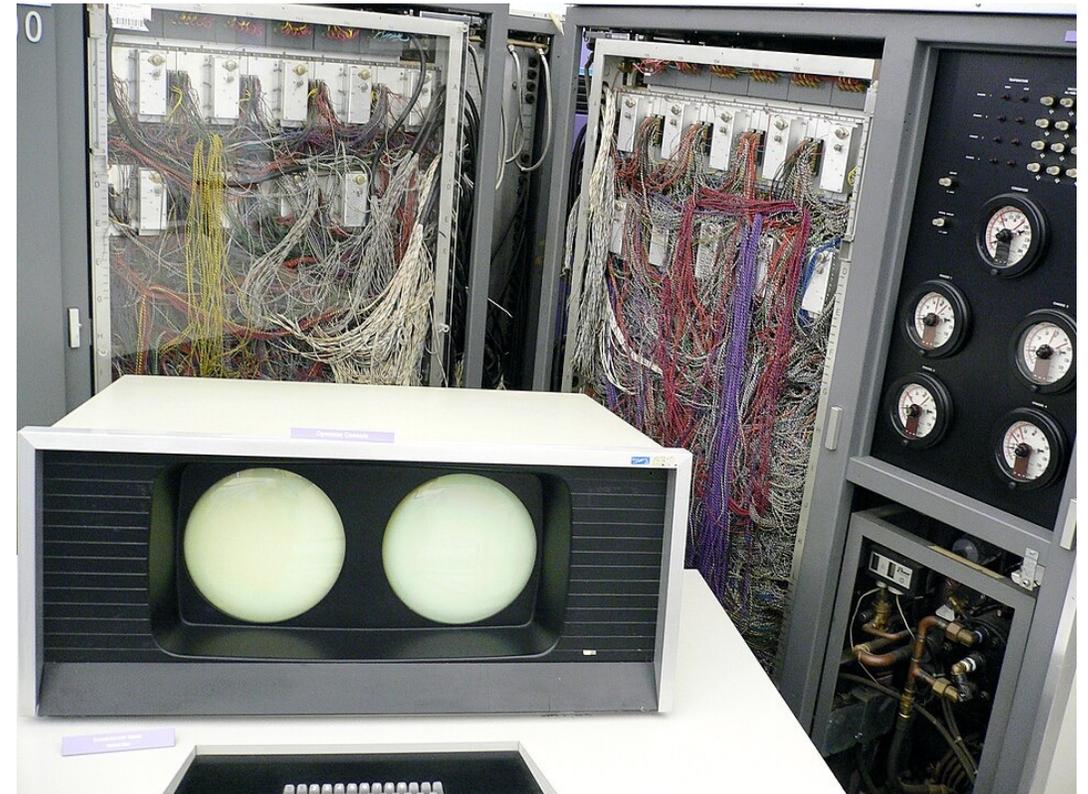
- 初期段階ではスーパーコンピュータ軍事目的で開発されてきた。
- **ENIAC**, 1943, 米国: 砲撃射表の計算を目的にペンシルバニア大学で1943~1946年に開発、1955年まで稼働した。
- **Colossus**, 1943, 英国: エニグマ等の暗号解読用に設計されたが、WWII後に破壊された。
- **Z3**, 1941?, ドイツ
- 米国、英国、ドイツでその他初期計算機が開発されてきた。



ENIAC: ピーク性能 0.00289 MIPS

# 初期のスーパーコンピュータ 2

- **IBM 704**, 1954, 米国
- **IBM 7090**, 1959 米国
- **LARC**, 1960, 米国
- **IBM 7030**, 1961, 米国
- **CDC 6600**, 1965, 米国 :  
CDC社においてシーモア・クレイにより設計され、当時のどのコンピュータより10倍高速であった。下はこの後継機
- **CDC 7600**, 1965, 米国



CDC 6600: ピーク性能1MIPS/3MFLOPS

# ベクトル機時代のスーパーコンピュータ

- **CDC Star-100**, 1974, 米国:  
最初のベクトル型計算機
- **Cray-1**, 1975, 米国:  
シーモア・クレイのクレイリサーチ社  
が開発した。最初の自動ベクトル  
化FORTRANコンパイラを備える
- **Cray X-MP**, 1983, 米国
- **Cray-2**, 1985, 米国
- **富士通 数値風洞(NWT)**,  
1993, 日本:  
CFD計算で $>100$ GFLOPS,  
1993~1996年で1位にランク



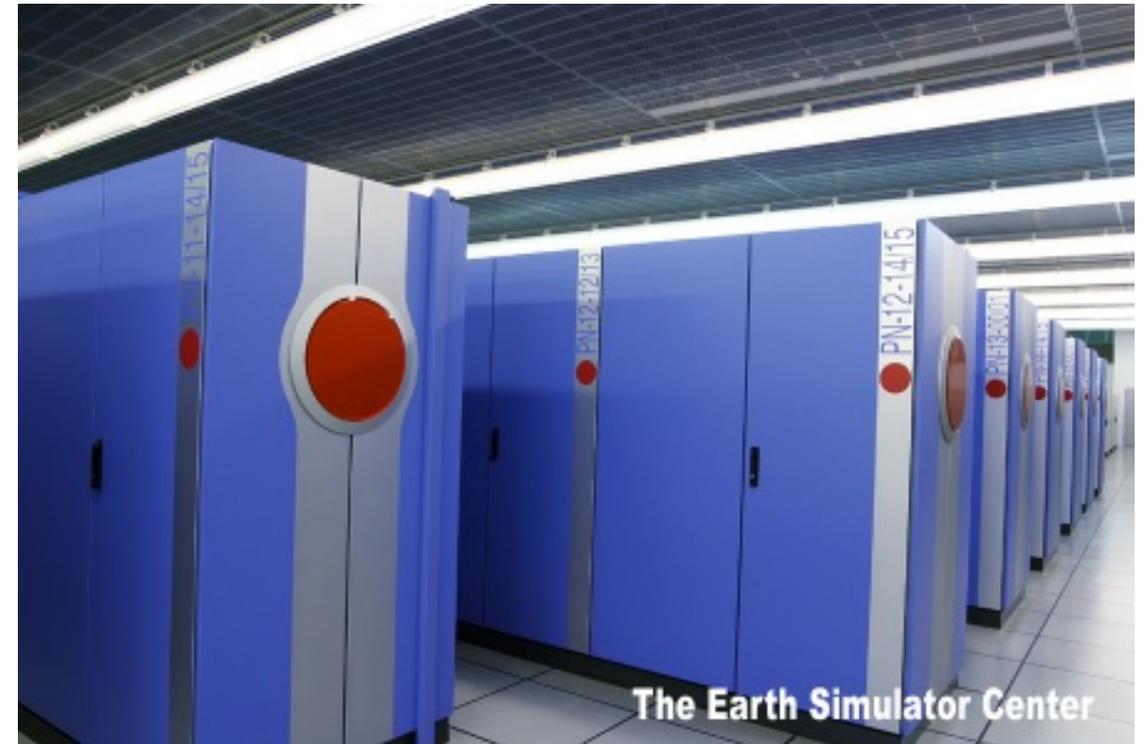
Cray-1: ピーク性能 80MPLOS(Vec), 72MFLOPS(Scal)



富士通 数値風洞: ピーク性能 236GFLOPS

# 最近のスーパーコンピュータ 1

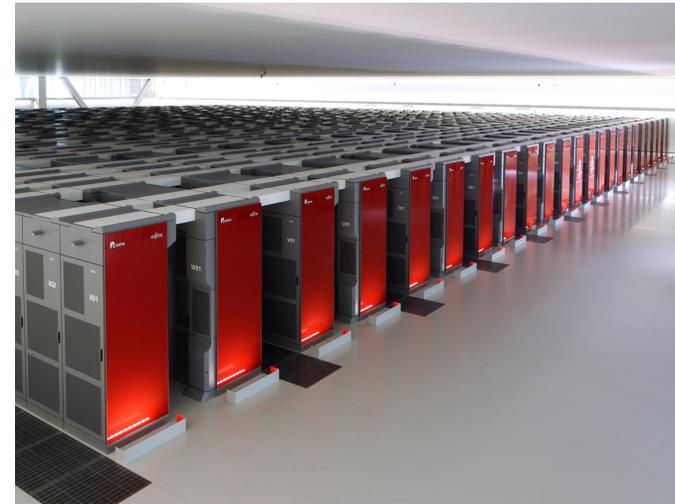
- Intel ASCI-Red, 1997, 米国
- IBM ASCI-White, 2000, 米国
- NEC 地球シミュレータ, 2002, 日本:  
NEC SX-6ベースのベクトル並列  
計算機
- IBM ASCI-Blue Gene/L,  
2005, 米国
- IBM Roadrunner, 2008, 米国
- Cray Jaguar, 2009, 米国



NEC 地球シミュレータ: ピーク性能 40TFLOPS

# 最近のスーパーコンピュータ 2

- **Tianhe-1A**, 2010, 中国
- **富士通 京コンピュータ**, 2011, 日本
- **IBM Sequoia**, 2012, 米国
- **Tianhe-2**, 2013, 中国
- **Sunway TaihuLight**, 2015, 中国
- **Summit**, 2018, 米国
- **富士通 富岳**, 2020, 日本
- **Frontier**, 2022, 米国:  
最初のエクサスケール計算機  
AMD製CPUとGPUを装備



富士通 京コンピュータ: ピーク性能 10.5PFLOPS



富岳: ピーク性能 537PFLOPS

# 現在のスーパーコンピュータの性能ランキング

- **TOP500**により認定される：第1位はスーパーコンピュータFrontier(米国); HPE Cray, AMD CPU + AMD GPU, 理論性能 1.7 EFLOPS

Rank	System	Cores	Rmax (PFlop/s)	Rpeak (PFlop/s)	Power (kW)
1	<b>Frontier</b> - HPE Cray EX235a, AMD Optimized 3rd Generation EPYC 64C 2GHz, AMD Instinct MI250X, Slingshot-11, HPE DOE/SC/Oak Ridge National Laboratory United States	8,699,904	1,194.00	1,679.82	22,703
2	<b>Aurora</b> - HPE Cray EX - Intel Exascale Compute Blade, Xeon CPU Max 9470 52C 2.4GHz, Intel Data Center GPU Max, Slingshot-11, Intel DOE/SC/Argonne National Laboratory United States	4,742,808	585.34	1,059.33	24,687
3	<b>Eagle</b> - Microsoft NDv5, Xeon Platinum 8480C 48C 2GHz, NVIDIA H100, NVIDIA Infiniband NDR, Microsoft Microsoft Azure United States	1,123,200	561.20	846.84	



- TOP500での性能測定は**HPL**(Linpack)：LU分解の一種であり、実アプリケーションの実効性能を反映していない → HPCGランキングの導入
- **HPCG**(Conjugate Gradient)はHPLを補完するランキング  
現在の第1位はスーパーコンピュータ富岳(日本)(TOP500では4位)
- **Green500**は消費電力で割った性能値：  
現在の第1位はHenri(米国)(TOP500では293位)

- ENIAC(1943)の時代は、プログラムは配線で与えられた。
- **FORTRAN**：科学技術計算用の高級言語(現在はFortranと書く)
- 1954年にIBMで考案され、1956年にIBM704用のコンパイラがリリースされた。
- 数値予報(NWP)、有限要素法(FEM)、数値流体力学(CFD)、地球物理学、計算物理学、結晶学、計算化学などの大規模計算において、スーパーコンピュータで使われている。
- 最新言語仕様はFortran2023。
- TIOBEで人気プログラミング言語の12位にランクイン。
- **C**や**C++**も次第に使われるようになってきた。
- FortranやC/C++の言語仕様自体は並列計算をサポートしていない。
- 並列計算には**MPI**ライブラリや、**OpenMP**拡張言語が併用されている。
- GPU計算には、**CUDA**や**OpenACC**が併用されている。

- lab2022上での計算手順
- CPU実行コンパイル:  
\$ mpiifort -qopenmp prog.F90
- 実行:  
\$ export ¥  
OMP\_NUM\_THREADS=16  
\$ mpirun -np 8 ./a.out
- GPU実行:  
!\$acc kernels~!\$acc end  
kernels
- コンパイル:  
\$ mpifort -acc -Minfo=accel ¥  
prog.F90
- 実行  
\$ mpirun -np 8 ./a.out

prog.F90

```
program main
  use mpi
  implicit none
  ...
  call MPI_Init(ierr)
  ...
  !$omp parallel do
  do i = ista, iend
  ...
  end do
  !$omp end parallel do
  ...
  call MPI_Finalize(ierr)
end program
```

# AI(Artificial Intelligence, 人工知能)

- **AIとは？**

計算機やソフトウェアの知能のこと(人間や生物の知能と対比)。インテリジェントマシンの開発研究を行う計算機科学の一分野

- **AI技術の利用:**

Web検索エンジン(Google Searchなど)、リコメンドシステム(YouTube、Amazonなど)、自然言語処理(Siri、Alexaなど)、自動運転(Waymo)、生成クリエイティブツール(ChatGPT、AIアート)、ゲーム(チェスや碁など)

- **最初のAI研究:**

アラン・チューリングが1950年に論文「計算する機械と知性」を公表:チューリングテスト

- **AIの定義:**

知的な機械、特に知的なコンピュータプログラムを作る科学・技術。人の知能を理解するためにコンピュータを使うことと似ている仕事であるが、生物学的に観測しうる方法に限定されるものではない(ジョン・マッカーシー,1956)。

- 人工知能の歴史は古代まで遡ると言われているが、ここでは省略。
- **前史**：現代のコンピュータ(ENIAC, Colossusなど)の開発。
- **人工知能の誕生(1943-1956)**：  
サイバーネティックスと初期のニューラルネットワーク、ゲームAI、チューリングテスト、記号的推論とLogic Theorist
- **第一次AIブーム(1956-1974)**：  
手段目標分析、自然言語、マイクロワールド
- **第二次AIブーム(1980-1987)**：  
エキスパートシステム、知識ベースシステム、知識工学
- **第三次AIブーム(2006-)**：  
ディープラーニング、ビッグデータ、  
実用化(Alpha Go、画像生成AI、ChatGPT)

# 人工知能関連のトピックと技法

- **AI:**  
ルールベースのエキスパートシステム、ロボティクス、コンピュータビジョン、自然言語処理、音声認識、対話システム、レコメンダーシステム、制約充足問題、知識表現と推論
- **機械学習:**  
人工ニューラルネットワーク、決定木、ランダムフォレスト、ロジスティック回帰、サポートベクトルマシン、ベイジアンネットワーク、アダブースト(Adaboost)、K-meansクラスタリング、主成分分析
- **ディープラーニング**(人工ニューラルネットワークの中):  
多層フィードフォワードネットワーク、畳み込みニューラルネットワーク(CNN)、再帰型ニューラルネットワーク(RNN)、敵対的生成ネットワーク(GAN)、オートエンコーダ、単語埋め込み(分散表現)
- **機械学習の分類:**教師あり学習(こちらがメイン)、教師なし学習、強化学習

- **Python:** スクリプト型高水準汎用プログラミング言語
- 1991年にガイド・ヴァン・ロッサムにより開発された。
- 人工知能や機械学習の分野で使われている。
- TIOBEでは人気プログラミング言語第1位となっている(2位はC、3位はC++)。
- Pythonはマルチスレッドやマルチプロセスなどの並列処理もサポート。
- ライブラリが充実している(GPU対応済みも多い)
  - 数値計算: NumPy、Scipy、Matplotlib
  - コンピュータビジョンや画像処理: OpenCV
  - 機械学習: Tensorflow、Keras、Pytorch、scikit-learn
- 今年度lab2022上でこれらの数値計算・人工知能計算の環境構築を行ない、試行計算を実施している。

## 畳み込みニューラルネットによる画像分類 : ConvolutionalNeuralNetwork.py

```
from tensorflow.keras.models import Sequential
from tensorflow.keras.layers import Dense, Flatten, MaxPooling2D, Conv2D, Dropout
from tensorflow.keras.optimizers import SGD
from tensorflow.keras.datasets import mnist

(x_train, y_train), (x_test, y_test) = mnist.load_data() # 訓練データと検証データの読み込み
model = Sequential([
    Conv2D(32, (5, 5), input_shape=(28, 28, 1), activation='relu'),
    MaxPooling2D(pool_size=(2,2)), ... ]) # ニューラルネットワークモデルの定義

model.compile(loss='categorical_crossentropy', optimizer=SGD(0.5), metrics=['accuracy']) # ネットワークのコンパイル
history = model.fit(x_train, y_train, epochs=5, batch_size=75, verbose=1) # 訓練
score = model.evaluate(x_test, y_test, verbose=1)
print('Accuracy = ', score[1])

for i in range(10):
    score = model.predict(x_test[i].reshape(1, 28, 28)) # 推論
    predicted = np.argmax(score)
    answer = np.argmax(y_test[i])
    print('Answer:' + str(answer) + ' Predicted:' + str(predicted))
```

## lab2022上でのCNNプログラム実行

訓練

```
$ python3.9 ConvolutionNeuralNetwork.py
Epoch 1/5
800/800 [=====] - 11s 13ms/step - loss: 0.2214 - accuracy: 0.9291
Epoch 2/5
800/800 [=====] - 10s 13ms/step - loss: 0.0579 - accuracy: 0.9823
...
Epoch 5/5
800/800 [=====] - 10s 13ms/step - loss: 0.0283 - accuracy: 0.9912
313/313 [=====] - 1s 3ms/step - loss: 0.0266 - accuracy: 0.9911
Accuracy = 0.991100013256073 # 正解率は99.1%
```

推論

```
1/1 [=====] - 0s 92ms/step
Answer:7 Predicted:7
1/1 [=====] - 0s 19ms/step
Answer:2 Predicted:2
...
1/1 [=====] - 0s 20ms/step
Answer:9 Predicted:9
```

- 手書き数字の画像データセットMNISTを学習してニューラルネットワークを訓練し(検証データでの正解率は99.1%)、推論を行うことができた。

# 人工知能(AI)と高性能計算(HPC)

- AIとHPCは基本的には異なる分野:
  - HPCはどちらかというハードウェア寄り(どんな大規模計算でも当てはまる)
  - AIは学習・推論するソフトウェア。
- HPCとAIの融合は不可欠。
- AIによる計算機シミュレーションの高度化:  
プラズマ物理学、重力波天体物理学、高エネルギー物理学、マルチメッセンジャー宇宙物理学、物質科学、非構造化データマネージメント、遺伝子データなど  
例：コストの大きい計算を機械学習に置き換えてスピードアップさせる。
- HPCによるAIアプリケーションの高度化:  
AIアプリケーションの並列実行により、ビッグデータの取り扱いが可能になる。
- ボトルネックはプログラミング言語の違い:  
HPCはFortran, C/C++、AIはPython(あるいはJulia)

# 本発表のまとめ

- 計算科学の見地から、HPC(高性能計算)とAI(人工知能)について述べた。
- 高性能計算分野
  - 現時点での最高速計算機はHPE Frontierスーパーコンピュータ(米国)：エクサフロップ( $10^{18}$ 回の浮動小数点演算/秒)
  - プログラミング言語: Fortran、C/C++
  - 並列化: MPI、OpenMP、OpenACC、CUDA(AMDはHIP)など
- 人工知能分野
  - 現在は第3次ブーム、ChatGPTなどで非常に注目を浴びている。
  - プログラミング技術: Python+Tensorflow?
- HPCとAIの連携研究プロジェクトが成果を挙げている。
- 今年度はlab2022におけるAIプログラミング環境の構築を行なった:  
Python+numpy, scipy, matplotlib, scikit-learn, matplotlib, jupyter, sympy, nltk, pandas, genism, cvxopt, tensorflow, torch