Faculty of Computer Application
B.Sc. (IT) (FinTech-IT in Finance)

- **Semester– 6**
- **Subject Code: 05FN0602**
- **Subject Name: Algorithmic Trading**
- **Objectives:**
  1. To introduce the fundamentals of algorithmic trading and its applications in financial markets.
  2. To equip students with the knowledge of trading strategies, indicators, and risk management.
  3. To develop practical skills using programming languages such as Python for algorithmic trading.
  4. To enable students to design, backtest, and optimize trading algorithms.
  5. To understand regulatory, ethical, and operational aspects of automated trading systems.

- **Prerequisites:** Basic knowledge of Python programming, Statistics, and Financial Markets.

Unit No	Topic Covered	No of Hours Required
1	**Introduction to Algorithmic Trading** • Basics of financial markets and trading systems • What is algorithmic trading? Advantages and challenges • Market types, order types, and trading lifecycle • Components of an algo trading system: data, strategy, execution	11
2	**Trading Strategies and Indicators** • Introduction to quantitative trading strategies • Technical indicators: RSI, MACD, Bollinger Bands, Moving Averages • Momentum vs Mean Reversion strategies • Portfolio construction and position sizing	11
3	**Backtesting and Performance Evaluation** • Historical data sourcing and preprocessing • Designing a backtesting engine in Python • Performance metrics: Sharpe ratio, drawdown, alpha/beta • Overfitting, look-ahead bias, and walk-forward testing	11

4	**Live Trading and Risk Management** • Setting up paper/live trading using broker APIs (Zerodha, Alpaca) • Algorithm execution: market/limit orders, slippage, latency • Risk controls: stop-loss, exposure limits, diversification • Regulatory framework and compliance in algo trading	**12**

Course Outcomes:

1. Understand the structure and function of algorithmic trading systems.
2. Apply technical indicators and develop quantitative trading strategies.
3. Perform backtesting and evaluate trading performance using Python.
4. Integrate and deploy trading bots with broker APIs for live execution.
5. Assess and implement risk management and regulatory practices in trading.

Course Outcomes – Program Outcomes Mapping Table:

	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PSO1	PSO2	PSO3
CO1	H	M		L					H	M	
CO2	M	H		M					H	M	M
CO3	M			H	M	H			H	M	M
CO4	H	H	M	L	H	M			M	L	
CO5	M	L		M		H	M		L	M	

Text Book:

- *Algorithmic Trading* by Ernest P. Chan, 2018

Reference Books:

- *Python for Algorithmic Trading* by Yves Hilpisch, 2020
- *Quantitative Trading* by Ernest P. Chan, 2019
- *Advances in Financial Machine Learning* by Marcos López de Prado, 2018

App & Web References:

- Zerodha Kite Connect API
- QuantInsti Quantra Modules

- Backtrader (Python library)
- Alpaca API Docs
- TradingView Strategy Tester

Syllabus Coverage from text /reference book & web/app reference:

Unit No	Chapter Numbers / Sources
1	Chan - Ch 1,2; QuantInsti Basics
2	Chan - Ch 4; Hilpisch Ch 3-4
3	Hilpisch Ch 6; Backtrader Docs
4	Alpaca/Zerodha Docs; De Prado Ch 5

PRACTICALS

	Numbers of Hours Required: 60	
1	• Create a basic order execution simulator • Import and visualize historical stock/crypto data using `yfinance` or APIs • Study different order types using simulation	
2	• Code RSI, MACD, Bollinger Bands in Python • Implement momentum vs mean-reversion strategies • Compare strategies on various stock datasets	
3	• Build a backtest function using vectorized logic • Evaluate strategies using Sharpe ratio, drawdown, CAGR • Perform walk-forward validation and tune parameters	
4	• Integrate trading bot using Zerodha/Alpaca API (paper trading) • Add stop-loss, profit targets, and logging features • Present a full working project: e.g., Intraday Equity Trader or Crypto Arbitrage Bot	